August 2017

Quantized Consensus by the Alternating Direction Method of Multipliers: Algorithms and Applications

Shengyu Zhu
Syracuse University

ABSTRACT

Collaborative in-network processing is a major tenet in the fields of control, signal processing, information theory, and computer science. Agents operating in a coordinated fashion can gain greater efficiency and operational capability than those perform solo missions. In many such applications the central task is to compute the global average of agents' data in a distributed manner. Much recent attention has been devoted to quantized consensus, where, due to practical constraints, only quantized communications are allowed between neighboring nodes in order to achieve the average consensus. This dissertation aims to develop efficient quantized consensus algorithms based on the alternating direction method of multipliers (ADMM) for networked applications, and in particular, consensus based detection in large scale sensor networks.

We study the effects of two commonly used uniform quantization schemes, dithered and deterministic quantizations, on an ADMM based distributed averaging algorithm. With dithered quantization, this algorithm yields linear convergence to the desired average in the mean sense with a bounded variance. When deterministic quantization is employed, the distributed ADMM either converges to a consensus or cycles with a finite period after a finite-time iteration. In the cyclic case, local quantized variables have the same sample mean over one period and hence each node can also reach a consensus. We then obtain an upper bound on the consensus error, which depends only on the quantization resolution and the average degree of the network. This is preferred in large scale networks where the range of agents' data and the size of network may be large.

Noticing that existing quantized consensus algorithms, including the above two, adopt infinite-bit quantizers unless a bound on agents' data is known *a priori*, we further develop an ADMM based quantized consensus algorithm using finite-bit bounded quantizers for possibly unbounded

agents' data. By picking a small enough ADMM step size, this algorithm can obtain the same consensus result as using the unbounded deterministic quantizer. We then apply this algorithm to distributed detection in connected sensor networks where each node can only exchange information with its direct neighbors. We establish that, with each node employing an identical one-bit quantizer for local information exchange, our approach achieves the optimal asymptotic performance of centralized detection. The statement is true under three different detection frameworks: the Bayesian criterion where the maximum *a posteriori* detector is optimal, the Neyman-Pearson criterion with a constant type-I error constraint, and the Neyman-Pearson criterion with an exponential type-I error constraint. The key to achieving optimal asymptotic performance is the use of a one-bit deterministic quantizer with controllable threshold that results in desired consensus error bounds.

QUANTIZED CONSENSUS BY THE ALTERNATING DIRECTION METHOD OF MULTIPLIERS: ALGORITHMS AND APPLICATIONS

By

Shengyu Zhu
B.E. in Electrical Engineering, Beijing Institute of Technology, 2010
M.S. in Mathematics, Syracuse University, 2016

DISSERTATION

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

Syracuse University
August 2017

ACKNOWLEDGMENTS

First and foremost, my deepest gratitude goes to my advisor, Dr. Biao Chen. I feel very fortunate and privileged to be his student. I always remember the time when my visa got checked and I could not catch the Fall semester. It was he who encouraged me and helped me postpone the enrollment date. During my Ph.D. years, I have learned a lot not only from his knowledge and wisdom but also from his working style and passion. I sincerely thank him for his guidance and continuous support. His profound thinking, generosity, and integrity will be an inspiring role model for my future career.

I would like to thank all my committee members for carefully reading my dissertation and many helpful suggestions. They are (in alphabetic order): Dr. Mustafa Cenk Gursoy, Dr. Yingbin Liang, Dr. Yan-Yeung Luk, Dr. Pramod Varshney, and Dr. Yi Wang.

Thanks to our lab mates and alumna: Xiaohu Shang, Yi Cao, Wei Liu, Fangfang Zhu, Kapil Borle, Pengfei Yang, and Yu Zhao, with whom I have shared a great time in my life. Special thanks to Ge and Pengfei for insightful discussions on research as well as their help in life.

I am also grateful to Professor Zhi-Quan Luo for visiting his group in Spring 2014. I would like to thank Ruoyu Sun, Mingyi Hong, Hung-Wei Tseng, Wei-Cheng Liao, and Yuan Yuan for their help during the stay. Special thanks to Mingyi for many helpful suggestions on my research. It was a great experience that I will always remember.

My final but everlasting gratitude goes to my parents, who have been a constant source of love and strength. It is their unconditional support that I have had the chances to pursue my Ph.D. degree.

I acknowledge the generous support of the National Science Foundation, the Air Force Research Laboratory, the Air Force Office of Scientific Research, and the Center for Advanced Systems and Engineering (CASE) at Syracuse University.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1
INTRODUCTION

Collaborative in-network processing is a major tenet in the fields of control, signal processing, information theory, and computer science. Agents operating in a coordinated fashion can gain greater efficiency and operational capability than those perform solo missions. A fundamental concern in such systems is the *consensus* problem which aims to reach an agreement among all agents. Of particular interest is the *distributed average consensus* which computes the global average of agents' data through only local computations and communications. Originating from distributed computation and decision-making [1, 2], distributed average consensus has arisen in various recent applications. For example, coordination for autonomous mobile agents [3–5] can often be formulated as a consensus average problem and is key to unmanned aerial vehicle (UAV) formation control and collision avoidance. Another application is in distributed hypothesis testing over a connected network [6, 7] where independently and identically distributed (i.i.d.) observations are collected across the network. Invariably, global log-likelihood ratio (LLR) is a sufficient statistic for all optimal detectors and with i.i.d. data, such a global LLR is simply the average of all local LLRs. Load balancing [8] is yet another example where task assignment across processors needs to equalize processing requirement and be completed in a timely manner.

Distributed averaging algorithms refer to iterative algorithms that aim to achieve the average consensus in a distributed manner. These algorithms are extremely attractive for large scale net-

works characterized by the lack of centralized access to information. They are also energy efficient and enhance the survivability of the networks compared with fusion center based processing. However, real networks can only allow messages with limited length to be transmitted between agents due to physical constraints such as limited bandwidth, sensor battery power, and computing resources. When a real value is sent from an agent to its neighbors, this value will be truncated or compressed and it is normally assumed that agents can reliably transmit only *quantized data*. The average consensus problem with this quantization constraint is referred to as *quantized consensus* in the literature [9], and iterative algorithms that can work with this constraint is called quantized consensus algorithms. This dissertation focuses on developing efficient quantized consensus algorithms for networked applications and in particular, consensus based detection in large scale sensor networks.

1.1 Literature Review

1.1.1 Distributed Averaging and Quantized Consensus Algorithms

There are three widely used methods for solving distributed average consensus problems. A classical approach is to update the state of each node with a weighted average of values from neighboring nodes [10–12]. The matrix, consisting of the weights associated with the edges, is chosen to be doubly stochastic to ensure convergence to the average. Another method is a gossip based algorithm, initially introduced in [1] for consensus problems and further studied in [9, 13, 14], among others. The third approach is to employ the alternating direction method of multipliers (ADMM) which is an iterative algorithm for solving convex problems and has received much attention recently (see [15] and references therein). The idea is to formulate the data average as the solution to a least-squares problem and manipulate the ADMM updates to derive a distributed algorithm [16–18]. Viewed from this point, applying distributed gradient descent or distributed (accelerated) proximal-gradient methods to the least-squares problem results in the classical method or some variants.

In the most ideal case where agents are able to send and receive real values with infinite precision, the three methods can all lead to the desired consensus at the average. When quantization is imposed, however, these methods do not directly apply. A well studied approach for quantized consensus is to use dithered quantizers which add noises to agents' variables before quantization [19]. By imposing certain conditions, the quantization error sequence becomes i.i.d. and is also independent of the input sequence. The classical approach and the gossip based algorithm then yield the almost sure consensus at a common but random quantization level with the expectation of the consensus value equal to the desired average [20–22]. To the best of our knowledge, there have been no existing results on the ADMM based method for quantized consensus. Nevertheless, since the quantization error of dithered quantizer is zero-mean and has a bounded variance, we can immediately extend the results in [17, 18] to quantized consensus (see Chapter 3.2). That is, the ADMM based method using dithered quantization leads to the consensus at the data average in the mean sense whose variance converges to a finite value.

Meanwhile, studies on distributed average consensus with deterministic quantizers have been scarcely reported. Deterministic quantization makes the problem much harder to deal with as the error terms caused by quantization no longer possess tractable statistical characteristics [20, 21]. The authors in [12] show that the classical approach, where a quantization rule that rounds the values down is adopted, converges to a consensus with an error from the average depending on the quantization resolution, the number of agents, the agents' data, and the updated weights of each agent. In [23], quantized consensus is formulated as a feedback control design problem for coding/decoding schemes. With an appropriate scaling function and carefully chosen control gain when some spectral properties of the Laplacian matrix of the underlying fixed undirected graph is known in advance, the proposed protocol with rounding quantizer is shown to achieve the exact average consensus asymptotically. A recent result of [24] indicates that this approach, with appropriate choices of the weights, reaches a quantized consensus close to the average in finite time or leads all agents' variables to cycle in a small neighborhood around the average; in the latter case, however, a consensus is not guaranteed. The gossip based algorithms in [22] and [9]

have similar results to those of the classical approach with probability one (or at least with high probability), where the randomness is from the random selection of the edge at each iteration. The ADMM based algorithms for deterministically quantized consensus, however, have not yet been explored.

1.1.2 Consensus Based Detection

An application of distributed average consensus lies in distributed detection in sensor networks, where the local data are LLRs of local observations and their average, called average LLR, is sufficient to achieve the optimal detection performance under broad conditions [25–31]. Different from canonical structures where there is a fusion center accessing information, consensus based detection deals with network inference problems in the absence of any fusion center [6, 7, 32–36]. Sensors iteratively exchange information with their neighbors to arrive at a consensus decision based on some consensus rules.

Practical channels, especially those employed in large scale sensor networks, are subject to strict bandwidth and resource limits, and again sensors are normally assumed to be able to reliably transmit only quantized data. Of particular interest is the extreme case where each sensor can only send one-bit information. Tsitsiklis established in [30] the optimality of identical likelihood ratio quantizers in such a setting for a canonical fusion network with communications allowed from the sensor to the fusion center (i.e., no consensus type iterations). The resulting decay rate is typically lower than the centralized one under the Neyman-Pearson or maximum *a posteoriori* (MAP) criteria. For the tandem network, it was shown in [31] that using a one-bit quantizer at each sensor can never achieve an exponential decay rate of the error probability under the MAP criterion. To the best of our knowledge, there is no asymptotic result on consensus based structures using one-bit quantizer at each node. As such, it is *a priori* unknown whether one-bit quantization in a general connected network that allows iterative communications can achieve exponentially decaying error probability and what would be the optimal exponent if exponentially vanishing error probability is feasible. Note that if nodes have perfect knowledge of global network topology, one

can construct schemes that utilize source coding ideas to attain the same optimal error exponent as in the centralized setting. This, however, is not realistic in most applications where nodes only have knowledge of their directly connected neighbors.

1.2 Motivations

1.2.1 Consensus Error Bound in Large Networks

We may roughly divide existing quantized consensus algorithms into two types according to their convergence results: one has asymptotic convergence to the exact average for each agent (e.g., [17,18,20–23]) and the other reaches a consensus within finite iterations at a cost of consensus error from the desired average (see [12, 22]). The second type of algorithms usually adopt deterministic quantizers and are preferred in situations where consensus needs reaching in finite time, e.g., a team of members may have to reach some agreement within finite time and can afford some suboptimal consensuses. Existing algorithms of the second type, however, have increasing consensus error bounds in the range of agents' data and the size of network, which may not be satisfactory enough in large scale networks. This motivates us to develop quantized consensus algorithms that are more favored in large networks.

1.2.2 Finite-bit Communications

We notice that the quantizers in most existing works are still of infinite bits since the quantizer output has unbounded range and infinite quantization levels. To our best knowledge, in order to employ finite-bit communications per iteration, all existing works using uniform quantizers assume that agents' data are bounded and the bound is known *a priori*, which is in general very restrictive and prohibits many networked applications. For example, consensus based detection has agents' data being local LLRS that can be arbitrarily large for common distributions, e.g., Gaussian distributions. Naively truncating the agents' data, however, has no consensus accuracy

guarantee. This is indeed the reason that there is no asymptotic characterization for consensus based detection using finite-bit data communications. Thus, quantized consensus algorithms using bounded quantizer for possibly unbounded data not only reduce the cost of data communications but also induce potential applications.

1.2.3 Asymptotically Optimal Detection in Sensor Networks

Sensor networks in today's applications can be very large. In the scenario of distributed detection, this implies the error exponent, assuming for now that error probabilities can decay exponentially in the size of networks, matters. Consensus based detection in sensor networks has been widely studied in [6, 7, 32–36] where exponential decay of error probabilities are indeed achieved in either online or offline settings. However, not only the error exponent is suboptimal to the centralized one in general, but also the data communicated among linked sensors are of infinite levels and hence of infinite bits. Simply using truncation to achieve finite-bit data communications do not have the same exponential decay results. Therefore, it is unknown if using finite-bit communications can lead to exponential decaying error probabilities, not to mention the optimal error exponent in centralized settings where all the observations are available for decision making.

On the other hand, the work of [30, 31] indicates that one-bit communications in general are not sufficient to achieve the optimal exponential decay of error probabilities in fusion center based schemes. This is not surprising as the information is highly compressed: each node send only one-bit information to the fusion center in parallel schemes or to the serial node in tandem schemes. In consensus based detection, while nodes can only communicate with their direct neighbors, the communications are allowed numerous times. In this sense, it might be possible that a quantized consensus approach can asymptotically achieve the optimal centralized error exponent using only finite-bit communications per iteration.

1.3 Contributions and Organization

Driven by the above facts, this dissertation aims to develop efficient quantized consensus algorithms for large scale networks and to apply them to distributed detection in connected sensor networks. We consider bidirectional connected networks with fixed topology and the quantizers used for data communications are uniform. Nodes can only communicate with their direct neighbors in a synchronous manner. The ADMM has been known to be an efficient algorithm for large scale optimizations and used in various applications such as regression and classification [15]. Moreover, the work of [37–39] validates the fast convergence of the ADMM and [17, 18] demonstrates the resilience of the ADMM to noise, link failures, etc. Also noticing that probabilistic quantization introduces additional randomness on the consensus result, making it difficult to achieve the optimal exponential decay, we therefore develop quantized consensus algorithms based on the ADMM and deterministic quantization schemes.

The contributions are summarized as follows:

- We study the effects of dithered and deterministic quantizations on an ADMM based distributed averaging algorithm. With probabilistic quantization, this algorithm yields linear convergence to the desired average in the mean sense with a bounded variance. When deterministic quantization is employed, it either converges to a consensus or cycles with a finite period after a finite-time iteration. In the cyclic case, local quantized variables have the same sample mean over one period and hence each node can also reach a consensus. We also obtain an upper bound on the consensus error which depends only on the quantization resolution and the average degree of the network. A two-stage algorithm is proposed which combines both probabilistic and deterministic quantizations. Simulations show that the two-stage algorithm, without picking small algorithm parameter, has consensus errors that are typically less than one quantization resolution for all connected networks where agents' data can be of large variance and magnitudes. These results have been reported in [40–42].

- Noticing the above quantizer needs infinite bits unless some prior information (e.g., bound

on agents' data) is known, we use finite-bit bounded quantizers to meet the communication constraint. We show that all the agent variables either converge to the same quantization level or cycle around the data average with the same sample mean over one period after a finite-time iteration. An error bound for the consensus value is obtained which turns out to be the same as that of using the unbounded deterministic quantizer, provided that the ADMM step size is small enough. We then study the effect of the algorithm parameter on our algorithms and propose a decreasing strategy for the parameter selection only using the number of agents and the number of edges in order to accelerate the algorithm with certain consensus accuracy guarantee. This part is from the work of [43, 44].

- We apply the ADMM based quantized consensus algorithm with finite-bit bounded quantization to distributed detection in connected sensor networks where each node can only exchange information with its direct neighbors. We establish that, by employing an identical one-bit quantizer for local information exchange, each node can achieve the optimal asymptotic performance of centralized detection; in particular, each node has its detection error probability decay exponentially with Chernoff information and Kullback-Leibler divergence as error exponents under the maximum *a posteriori* (MAP) criterion and Neyman-Pearson criterion with constant constraint, respectively. In addition, we examine non-asymptotic performance of the proposed approach and show that the type-I and type-II error probabilities at each node can be made arbitrarily close to the centralized ones simultaneously when a continuity condition is satisfied. These results are based on the work of [45, 46].

The rest of this dissertation is organized as follows. Chapter 2 reviews the application of the ADMM to distributed average consensus. Based on this distributed averaging algorithm, we develop quantized consensus algorithms using unbounded uniform quantizers in Chapter 3 and using finite-bit uniform quantizers in Chapter 4. In Chapter 5, we apply the proposed algorithm to distributed detection where only one-bit communication between linked nodes at each iteration is allowed. Chapter 6 concludes the dissertation and discusses several future research directions.

1.4 Notations

We use two definitions of rate of convergence for an iterative algorithm. A sequence $\boldsymbol{x}^k$, where the superscript k stands for time index, is said to converge *Q-linearly* to a point $\boldsymbol{x}^*$ if there exists a number $\upsilon \in (0,1)$ such that $\lim_{k\to\infty} \frac{\|\boldsymbol{x}^{k+1}-\boldsymbol{x}^*\|}{\|\boldsymbol{x}^k-\boldsymbol{x}^*\|} = \upsilon$ with $\|\cdot\|$ being a vector norm. A sequence $\boldsymbol{y}^k$ is said to converge *R-linearly* to $\boldsymbol{y}^*$ if for all k, $\|\boldsymbol{y}^k - \boldsymbol{y}^*\| \leq \|\boldsymbol{x}^k - \boldsymbol{x}^*\|$ where $\boldsymbol{x}^k$ converges Q-linearly to $\boldsymbol{x}^*$.

We use $\mathbf{0}$ (without subscript) to denote the all-zero column vector whose dimension can be decided from the context. $\mathbf{1}_K$ is the K-dimensional all-one column vector; $\mathbf{0}_K$ and $\boldsymbol{I}_K$ are the $K \times K$ all-zero and identity matrices, respectively. Notation $\otimes$ denotes the Kronecker product and $\|\boldsymbol{x}\|_2$ denotes the Euclidean norm of a vector $\boldsymbol{x}$. For $x \in \mathbb{R}$, $\lceil x \rceil$ is the ceiling function on x, i.e., the smallest integer that is greater than or equal to x. Given a positive semidefinite matrix $\boldsymbol{G}$ with proper dimensions, the $\boldsymbol{G}$-norm of $\boldsymbol{x}$ is $\|\boldsymbol{x}\|_{\boldsymbol{G}} = \sqrt{\boldsymbol{x}^T\boldsymbol{G}\boldsymbol{x}}$. For a real symmetric matrix $\boldsymbol{L}_{n\times n}$, denote its eigenvalues in the ascending order as $\lambda_1(\boldsymbol{L}) \leq \lambda_2(\boldsymbol{L}) \leq \cdots \leq \lambda_n(\boldsymbol{L})$. For any matrix $\boldsymbol{M}$, $\mathcal{C}(\boldsymbol{M})$ denotes its column space.

CHAPTER 2
REVIEW OF THE ADMM FOR DISTRIBUTED AVERAGE CONSENSUS

This chapter briefly reviews consensus ADMM (CADMM) for distributed average consensus where agents can communicate real data of infinite precision, aiming to provide a good understanding of how the ADMM works and performs in the distributed setting. We start with the network model that is used throughout this dissertation.

2.1 Network Model

Consider a connected network of n agents which are bidirectionally connected by m edges (and thus $2m$ arcs). We describe this network as a symmetric directed graph $\mathcal{G}_d = \{\mathcal{V}, \mathcal{A}\}$ or an undirected graph $\mathcal{G}_u = \{\mathcal{V}, \mathcal{E}\}$, where $\mathcal{V}$ is the set of vertices with cardinality $|\mathcal{V}| = n$, $\mathcal{A}$ is the set of arcs with $|\mathcal{A}| = 2m$ and $\mathcal{E}$ is the set of edges with $|\mathcal{E}| = m$. Define the oriented incidence matrix $\boldsymbol{M}_- \in \mathbb{R}^{n\times 2m}$ with respect to $\mathcal{G}_d$ as follows: $[\boldsymbol{M}_-]_{i,l} = 1$ if the lth arc leaves agent i, $[\boldsymbol{M}_-]_{i,l} = -1$ if the lth enters agent i, and $[\boldsymbol{M}_-]_{i,l} = 0$ otherwise. The unoriented incidence matrix $\boldsymbol{M}_+ \in \mathbb{R}^{n\times 2m}$ is defined by setting $[\boldsymbol{M}_+]_{i,l} = |[\boldsymbol{M}_-]_{i,l}|$. Denote $\mathcal{N}_i = \{j : (i,j) \in \mathcal{A}\}$ as the set of neighbors of agent i. Further define $\boldsymbol{L}_- = \frac{1}{2}\boldsymbol{M}_-\boldsymbol{M}_-^T$ and

$\boldsymbol{L}_+ = \frac{1}{2}\boldsymbol{M}_+\boldsymbol{M}_+^T$ which are respectively the signed and signless Laplacian matrices with respect to $\mathcal{G}_u$. Then $\boldsymbol{W} = \frac{1}{2}(\boldsymbol{L}_- + \boldsymbol{L}_+) = \text{diag}\{|\mathcal{N}_1|, |\mathcal{N}_2|, \ldots, |\mathcal{N}_n|\}$ is the degree matrix related to $\mathcal{G}_u$, i.e., a diagonal matrix with the (i,i)-th entry being $|\mathcal{N}_i|$ and other entries being 0.

The following lemma states useful properties about the connected network.

Lemma 2.1 ([47,48]). *Given a connected network, we have that*

a) $\boldsymbol{L}_-$ *is positive semidefinite and* $0 = \lambda_1(\boldsymbol{L}_-) < \lambda_2(\boldsymbol{L}_-) \leq \lambda_3(\boldsymbol{L}_-) \leq \cdots \leq \lambda_n(\boldsymbol{L}_-)$. $\boldsymbol{L}_-\boldsymbol{b} = \boldsymbol{0}$ *if and only if* $\boldsymbol{b} \in \mathcal{C}(\boldsymbol{1}_n)$ *with* $\boldsymbol{1}_n$ *being the all-one vector of dimension* n .

b) $\boldsymbol{L}_+$ *is positive semidefinite and* $\lambda_n(\boldsymbol{L}_+) > 0$.

c) $\mathcal{C}(\boldsymbol{M}_-) = \mathcal{C}(\boldsymbol{L}_-)$. *For every* $\boldsymbol{\alpha} \in \mathcal{C}(\boldsymbol{L}_-)$, *there exists a unique* $\boldsymbol{\beta} \in \mathcal{C}(\boldsymbol{M}_-^T)$ *such that* $\boldsymbol{\alpha} = \boldsymbol{M}_-\boldsymbol{\beta}$.

2.2 The ADMM and Consensus ADMM

The ADMM applies in general to the convex optimization problem in the form of

$$\begin{aligned} \underset{\boldsymbol{y}_1, \boldsymbol{y}_2}{\text{minimize}} \quad & g_1(\boldsymbol{y}_1) + g_2(\boldsymbol{y}_2) \\ \text{subject to} \quad & \boldsymbol{C}_1\boldsymbol{y}_1 + \boldsymbol{C}_2\boldsymbol{y}_2 = \boldsymbol{c}, \end{aligned} \tag{2.1}$$

where $\boldsymbol{y}_1$ and $\boldsymbol{y}_2$ are optimization variables, g_1 and g_2 are convex functions, and $\boldsymbol{C}_1\boldsymbol{y}_1 + \boldsymbol{C}_2\boldsymbol{y}_2 = \boldsymbol{c}$ is a linear constraint on $\boldsymbol{y}_1$ and $\boldsymbol{y}_2$. The ADMM solves a sequence of subproblems involving g_1 and g_2 one at a time and iterate to converge under mild conditions, e.g., g_1 and g_2 are proper closed convex functions and the Lagrangian of (2.1) has a saddle point [15].

CADMM is obtained by applying the ADMM to minimizing the sum of several convex functions in a distributed manner. Let $f_i : \mathbb{R}^d \to \mathbb{R}$, where d is a positive integer, denote a convex local objective function that is only known to agent i. In many cooperative applications, the goal is to

use only local computation and communication to find

$$\tilde{\boldsymbol{x}}^* = \arg\min_{\tilde{\boldsymbol{x}}} \sum_{i=1}^{n} f_i(\tilde{\boldsymbol{x}}). \tag{2.2}$$

To obtain the CADMM, we rewrite the above problem in the ADMM form as:

$$\begin{aligned} \underset{\{\boldsymbol{x}_i\},\{\boldsymbol{z}_{ij}\}}{\text{minimize}} \quad & \sum_{i=1}^{n} f_i(\boldsymbol{x}_i) \\ \text{subject to} \quad & \boldsymbol{x}_i = \boldsymbol{z}_{ij}, \boldsymbol{x}_j = \boldsymbol{z}_{ij}, \forall (i,j) \in \mathcal{A}, \end{aligned} \tag{2.3}$$

where $\boldsymbol{x}_i \in \mathbb{R}^d$ is the local copy of the common optimization variable $\tilde{\boldsymbol{x}}$ at agent i and $\boldsymbol{z}_{ij} \in \mathbb{R}^d$ is an auxiliary variable imposing the consensus constraint on neighboring agents i and j. This consensus constraint ensures the consensus to be achieved over the entire network, i.e., $\boldsymbol{x}_i = \boldsymbol{x}_j$ for all $i, j \in \mathcal{V}$, which in turn guarantees that (2.3) is equivalent to (2.2). Further define $\boldsymbol{x} \in \mathbb{R}^{nd}$ as a vector concatenating all $\boldsymbol{x}_i$, $\boldsymbol{z} \in \mathbb{R}^{2md}$ as a vector concatenating all $\boldsymbol{z}_{ij}$, and $f(\boldsymbol{x}) = \sum_{i=1}^{n} f_i(\boldsymbol{x}_i)$. Then (2.3) can be written in a matrix form as

$$\begin{aligned} \underset{x,z}{\text{minimize}} \quad & f(\boldsymbol{x}) + g(\boldsymbol{z}) \\ \text{subject to} \quad & \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{z} = \boldsymbol{0}, \end{aligned} \tag{2.4}$$

where $g(\boldsymbol{z}) = 0$, and $\boldsymbol{0}$ is a column vector with proper dimensions and all entries being 0. Here $\boldsymbol{B} = [-\boldsymbol{I}_{2md}; -\boldsymbol{I}_{2md}]$ with $\boldsymbol{I}_{2md}$ being a $2md \times 2md$ identity matrix and $\boldsymbol{A} = [\boldsymbol{A}_1; \boldsymbol{A}_2]$ with $\boldsymbol{A}_1, \boldsymbol{A}_2 \in \mathbb{R}^{2md \times nd}$. If $(i,j) \in \mathcal{A}$ and z_{ij} is the qth block of $\boldsymbol{z}$, then the (q,i)th block of $\boldsymbol{A}_1$ and the (q,j)th block of $\boldsymbol{A}_2$ are $\boldsymbol{I}_d$; otherwise the corresponding entries are $\boldsymbol{0}_d$.

With this matrix form, we are ready to write out the ADMM updates. The augmented Lagrangian of (2.4) is

$$L_\rho(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{\lambda}) = f(\boldsymbol{x}) + \langle \boldsymbol{\lambda}, \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{z} \rangle + \frac{\rho}{2} \|\boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{z}\|_2^2, \tag{2.5}$$

where $\boldsymbol{\lambda} = [\boldsymbol{\beta}; \boldsymbol{\gamma}]$ with $\boldsymbol{\beta}, \boldsymbol{\gamma} \in \mathbb{R}^{2md}$ is the Lagrange multiplier and ρ is a positive algorithm

parameter. At iteration $k+1$, the ADMM first obtains $\boldsymbol{x}^{k+1}$ by minimizing $L_\rho(\boldsymbol{x}, \boldsymbol{z}^k, \boldsymbol{\lambda}^k)$, then calculates $\boldsymbol{z}^{k+1}$ by minimizing $L_\rho(\boldsymbol{x}^{k+1}, \boldsymbol{z}, \boldsymbol{\lambda}^k)$ and finally updates $\boldsymbol{\lambda}^{k+1}$ using $\boldsymbol{x}^{k+1}$ and $\boldsymbol{z}^{k+1}$. The updates are

$$\begin{aligned} &\boldsymbol{x}\text{-update }:\partial f(\boldsymbol{x}^{k+1}) + \boldsymbol{A}^T\boldsymbol{\lambda}^k + \rho\boldsymbol{A}^T(\boldsymbol{A}\boldsymbol{x}^{k+1} + \boldsymbol{B}\boldsymbol{z}^k) = \boldsymbol{0}, \\ &\boldsymbol{z}\text{-update : } \boldsymbol{B}^T\boldsymbol{\lambda}^k + \rho\boldsymbol{B}^T(\boldsymbol{A}\boldsymbol{x}^{k+1} + \boldsymbol{B}\boldsymbol{z}^{k+1}) = \boldsymbol{0}, \\ &\boldsymbol{\lambda}\text{-update : } \boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k - \rho(\boldsymbol{A}\boldsymbol{x}^{k+1} + \boldsymbol{B}\boldsymbol{z}^{k+1}) = \boldsymbol{0}, \end{aligned} \tag{2.6}$$

where $\partial f(\boldsymbol{x}^{k+1})$ is a subgradient of f at $\boldsymbol{x}^{k+1}$. The $\boldsymbol{x}$-update and $\boldsymbol{z}$-update can also be viewed as proximal updates; see [49].

A nice property of the ADMM, known as *global convergence*, states that the sequence $(\boldsymbol{x}^k, \boldsymbol{z}^k, \boldsymbol{\lambda}^k)$ generated by (2.6) has a single limit point $(\boldsymbol{x}^*, \boldsymbol{z}^*, \boldsymbol{\lambda}^*)$ which is a primal-dual solution to (2.5), as stated in Lemma 2.2.

Lemma 2.2 (Global convergence of the ADMM [15,37,39])**.** *Assume that local objective functions f_i are proper closed convex functions and that the minimum of (2.2) is attainable. For any initial values $\boldsymbol{x}^0 \in \mathbb{R}^{nd}$, $\boldsymbol{z}^0 \in \mathbb{R}^{2md}$ and $\boldsymbol{\lambda}^0 \in \mathbb{R}^{4md}$, the update in (2.6) yields that*

$$\boldsymbol{x}^k \to \boldsymbol{x}^*,\ \boldsymbol{z}^k \to \boldsymbol{z}^*,\ \textit{and } \boldsymbol{\lambda}^k \to \boldsymbol{\lambda}^* \textit{ as } k \to \infty,$$

where $(\boldsymbol{x}^*, \boldsymbol{z}^*, \boldsymbol{\lambda}^*)$ is a primal-dual solution to (2.5).

While (2.6) provides an efficient centralized algorithm to solve (2.2), it is not clear whether (2.6) can be carried out in a distributed manner, i.e., data exchanges only occur within neighboring nodes. Interestingly, Lemma 2.2 states that convergence for the ADMM is guaranteed regardless of initial values $\boldsymbol{x}^0, \boldsymbol{z}^0$ and $\boldsymbol{\lambda}^0$; there indeed exist initial values that decentralize (2.6). Initialize $\boldsymbol{\beta}^0 = -\boldsymbol{\gamma}^0$ and $\boldsymbol{z}^0 = \frac{1}{2}(\boldsymbol{M}_+^T \otimes \boldsymbol{I}_d)\boldsymbol{x}^0$ where $\otimes$ denotes the Kronecker product. As shown in [16,50],

the updates in (2.6) lead to the following iterative updates:

$$\begin{aligned}
&\partial f_i(\boldsymbol{x}_i^{k+1}) + 2\rho|\mathcal{N}_i|\boldsymbol{x}_i^{k+1} - \left(\rho|\mathcal{N}_i|\boldsymbol{x}_i^k + \rho\sum_{j\in\mathcal{N}_i}\boldsymbol{x}_j^k - \boldsymbol{\alpha}_i^k\right) = \mathbf{0},\\
&\boldsymbol{\alpha}_i^{k+1} = \boldsymbol{\alpha}_i^k + \rho\left(|\mathcal{N}_i|\boldsymbol{x}_i^{k+1} - \sum_{j\in\mathcal{N}_i}\boldsymbol{x}_j^{k+1}\right),
\end{aligned} \tag{2.7}$$

at agent i, where $\boldsymbol{\alpha}_i^k \in \mathbb{R}^d$ is the local Lagrangian multiplier of agent i. The above updates are fully decentralized as the update of $\boldsymbol{x}_i^{k+1}$ and $\boldsymbol{\alpha}_i^{k+1}$ only relies on local and neighboring information. We refer to (2.7) as CADMM.

The following theorem states the convergence of CADMM, which follows directly from global convergence of the ADMM.

Lemma 2.3 (Convergence of CADMM [15, 37])**.** *Assume that local objective functions f_i are proper closed convex functions and that the minimum of (2.2) is attainable. Then CADMM is guaranteed to converge for any $\boldsymbol{x}_i^0 \in \mathbb{R}^d$, $[\boldsymbol{\alpha}_1^0; \boldsymbol{\alpha}_2^0; \cdots; \boldsymbol{\alpha}_n^0] \in \mathcal{C}(\boldsymbol{L}_- \otimes \boldsymbol{I}_d)$, and $\rho > 0$:*

$$\lim_{k\to\infty}\boldsymbol{x}_i^k = \tilde{\boldsymbol{x}}^* \text{ and } \lim_{k\to\infty}\boldsymbol{\alpha}_i^k = \boldsymbol{\alpha}_i^*, i = 1, 2, \ldots, n,$$

where $\boldsymbol{x}_1^* = \cdots = \boldsymbol{x}_n^* \triangleq \tilde{\boldsymbol{x}}^*$ and $\boldsymbol{\alpha}_i^*$ are a pair of primal and dual solutions to (2.3), and $\tilde{\boldsymbol{x}}^*$ is optimal to (2.2).

The ADMM and CADMM are also shown to converge linearly when certain convexity assumptions on the objective functions f_i's are satisfied; see, e.g., [38, 39, 50]. As this dissertation focuses on distributed average consensus, we only consider the convergence rate result when CADMM is applied to distributed average consensus. This is presented in the next section.

2.3 CADMM for Distributed Average Consensus

Denote by $r_i \in \mathbb{R}$ the measurement at agent i, $\boldsymbol{r}$ the vector that concatenates all r_i, and $\bar{r} = \frac{1}{n}\sum_{i=1}^n r_i$ the global average. Distributed average consensus then can be solved using CADMM

by identifying that the average is the unique solution to a least-squares problem, i.e.,

$$\bar{r} = \arg\min_{\tilde{x}} \frac{1}{2} \sum_{i=1}^{n} (\tilde{x} - r_i)^2.$$

Therefore, we have the CADMM updates for distributed averaging by plugging in $f_i(\tilde{x}) = \frac{1}{2}(\tilde{x} - r_i)^2$:

$$\begin{aligned} x_i^{k+1} &= \frac{1}{1 + 2\rho|\mathcal{N}_i|} \left(\rho|\mathcal{N}_i| x_i^k + \rho \sum_{j \in \mathcal{N}_i} x_j^k - \alpha_i^k + r_i \right), \\ \alpha_i^{k+1} &= \alpha_i^k + \rho \left(|\mathcal{N}_i| x_i^k - \sum_{j \in \mathcal{N}_i} x_j^k \right). \end{aligned} \tag{2.8}$$

For ease of presentation, we will use 'CADMM' to refer to the above updates and 'original CADMM' for the updates in (2.7) where local objectives are general convex functions. We can further write (2.8) in a matrix form as

$$\begin{aligned} \boldsymbol{x}^{k+1} &= (\boldsymbol{I}_n + 2\rho \boldsymbol{W})^{-1} (\rho \boldsymbol{L}_+ \boldsymbol{x}^k - \boldsymbol{\alpha}^k + \boldsymbol{r}), \\ \boldsymbol{\alpha}^{k+1} &= \boldsymbol{\alpha}^k + \rho \boldsymbol{L}_- \boldsymbol{x}^{k+1}, \end{aligned}$$

or more compactly,

$$\boldsymbol{s}^{k+1} = \boldsymbol{D} \boldsymbol{s}^k, \tag{2.9}$$

where $\boldsymbol{s}^k = \left[\boldsymbol{x}^k; \boldsymbol{\alpha}^k; \boldsymbol{r} \right]$, $\boldsymbol{D}_0 = (\boldsymbol{I}_n + 2\rho \boldsymbol{W})^{-1}$, and

$$\boldsymbol{D} = \begin{bmatrix} \rho \boldsymbol{D}_0 \boldsymbol{L}_+ & -\boldsymbol{D}_0 & \boldsymbol{D}_0 \\ \rho^2 \boldsymbol{L}_- \boldsymbol{D}_0 \boldsymbol{L}_+ & \boldsymbol{I}_n - \rho \boldsymbol{L}_- \boldsymbol{D}_0 & \rho \boldsymbol{L}_- \boldsymbol{D}_0 \\ \boldsymbol{0}_n & \boldsymbol{0}_n & \boldsymbol{I}_n \end{bmatrix}. \tag{2.10}$$

Following (2.9), we can write $\boldsymbol{s}^k$ as

$$\boldsymbol{s}^k = \boldsymbol{D}^k \boldsymbol{s}^0.$$

It is thus interesting to investigate how $\boldsymbol{D}^k$ behaves as $k \to \infty$. From (2.10), a logical approach is to study $\boldsymbol{D}^k$ through the structures of $\boldsymbol{L}_-, \boldsymbol{L}_+$ and $\boldsymbol{W}$; fortunately, the convergence property of CADMM provides a simple argument to obtain a rough estimate of $\boldsymbol{D}^\infty$, which, nevertheless, is good enough for our purpose. Note that we also have $\boldsymbol{D}^* = \boldsymbol{D}^\infty$ and $\boldsymbol{s}^* = [\boldsymbol{x}^*; \boldsymbol{\alpha}^*; \boldsymbol{r}] = [\boldsymbol{x}^\infty; \boldsymbol{\alpha}^\infty; \boldsymbol{r}] = \boldsymbol{s}^\infty$ as the optimum due to the convergence of CADMM. The result is given below

Theorem 2.1. *Consider $\boldsymbol{D}$ defined in (2.10). Then*

$$\boldsymbol{D}^* = \begin{bmatrix} \boldsymbol{D}_{11} & \boldsymbol{D}_{12} & \boldsymbol{D}_{13} \\ \boldsymbol{D}_{21} & \boldsymbol{D}_{22} & \boldsymbol{D}_{23} \\ \boldsymbol{D}_{31} & \boldsymbol{D}_{32} & \boldsymbol{D}_{33} \end{bmatrix} = \begin{bmatrix} \boldsymbol{0}_n & \boldsymbol{a}_1 \boldsymbol{1}_n^T & \frac{1}{n} \boldsymbol{1}_n \boldsymbol{1}_n^T \\ \boldsymbol{0}_n & \boldsymbol{a}_2 \boldsymbol{1}_n^T & \boldsymbol{I}_n - \frac{1}{n} \boldsymbol{1}_n \boldsymbol{1}_n^T \\ \boldsymbol{0}_n & \boldsymbol{0}_n & \boldsymbol{I}_n \end{bmatrix}$$

for fixed $\boldsymbol{a}_1, \boldsymbol{a}_2 \in \mathbb{R}^n$.

Proof. By Lemma 2.3, we have for any $\boldsymbol{s}^0$ that satisfies the initialization condition,

$$\boldsymbol{s}^\infty = \begin{bmatrix} \boldsymbol{x}^\infty \\ \boldsymbol{\alpha}^\infty \\ \boldsymbol{r}^\infty \end{bmatrix} = \begin{bmatrix} \boldsymbol{x}^* \\ \boldsymbol{\alpha}^* \\ \boldsymbol{r}^* \end{bmatrix} = \begin{bmatrix} \boldsymbol{1}_n \bar{r} \\ \boldsymbol{r} - \boldsymbol{1}_n \bar{r} \\ \boldsymbol{r} \end{bmatrix}.$$

Recall that $\boldsymbol{s}^\infty = \boldsymbol{D}^\infty \boldsymbol{s}^0$. If we fix $\boldsymbol{\alpha}^0$ and $\boldsymbol{r}^0$, global convergence implies that $\boldsymbol{s}^\infty = \boldsymbol{s}^*$ regardless of the initial value $\boldsymbol{x}^0$. Thus $\boldsymbol{D}_{i1} = \boldsymbol{0}_n, i = 1, 2, 3$. Similarly, fixing $\boldsymbol{x}^0$ and $\boldsymbol{r}^0$, we must have $\boldsymbol{D}_{12}\boldsymbol{\alpha}^0 = \boldsymbol{D}_{22}\boldsymbol{\alpha}^0 = \boldsymbol{0}$. Since $\boldsymbol{\alpha}^0$ is initialized in the column space of $\boldsymbol{M}_- \boldsymbol{M}_-^T = 2\boldsymbol{L}_-$ where $\boldsymbol{L}_-$ is the signed Laplacian matrix of a connected undirected graph, $\boldsymbol{D}_{12}$ and $\boldsymbol{D}_{22}$ must be respectively the products of some vectors $\boldsymbol{a}_1$ and $\boldsymbol{a}_2$ in $\mathbb{R}^n$ multiplying $\boldsymbol{1}_n^T$ such that $\boldsymbol{D}_{12}\boldsymbol{L}_- = \boldsymbol{D}_{22}\boldsymbol{L}_- = \boldsymbol{0}$. Knowing the form of $\boldsymbol{D}_{j1}$ and $\boldsymbol{D}_{j2}$ with $j = 1, 2$, we see that $\boldsymbol{x}^\infty$ and $\boldsymbol{\alpha}^\infty$ only depend on $\boldsymbol{r}^0 = \boldsymbol{r}$. Together with the facts that $\boldsymbol{x}^\infty = \boldsymbol{x}^*$ has each entry of itself reaching the data average $\bar{r} = \frac{1}{n}\boldsymbol{r}^T \boldsymbol{1}_n$ and that $\boldsymbol{\alpha}^\infty = \boldsymbol{r} - \boldsymbol{1}_n \bar{r}$ for any $\boldsymbol{r}$, we validate $\boldsymbol{D}_{13}$ and $\boldsymbol{D}_{23}$ as given in the theorem. The remaining blocks, $\boldsymbol{D}_{32}$ and $\boldsymbol{D}_{33}$, follow directly from the matrix multiplication. □

Given convergence, we now turn our attention to the rate of convergence of CADMM. Recent work of [38, 39] has established the linear convergence of the ADMM. Unfortunately, their results do not apply to CADMM as their conditions are not satisfied here. In [38], the step size of the dual variable update, i.e., ρ in the $\boldsymbol{\lambda}$-update of (2.6), need be sufficiently small while CADMM has a fixed step size ρ that can be any positive number (see Remark 3.3 for further discussion on the choice of ρ). The linear convergence in [39] is established provided that either $g(\boldsymbol{z})$ is strongly convex or $\boldsymbol{B}$ is full row-rank in (2.3). In our formulation, however, $g(\boldsymbol{z}) = 0$ is not strongly convex and $\boldsymbol{B} = [-\boldsymbol{I}_{2m}; -\boldsymbol{I}_{2m}]$ is row-rank deficient. Fortunately, [50, Theorem 1] characterizes the convergence rate of a vector concatenating $\boldsymbol{z}$ and $\boldsymbol{\beta}$, which can be used to derive the convergence rate about $\boldsymbol{s}^k$. Before stating this result in Theorem 2.2, we futher define

$$\boldsymbol{u}^k = \begin{bmatrix} \boldsymbol{z}^k \\ \boldsymbol{\beta}^k \end{bmatrix} \text{ and } \boldsymbol{G} = \begin{bmatrix} \rho \boldsymbol{I}_{2m} & \boldsymbol{0}_{2m} \\ \boldsymbol{0}_{2m} & \frac{1}{\rho}\boldsymbol{I}_{2m} \end{bmatrix}. \tag{2.11}$$

Theorem 2.2 (Linear convergence of CADMM for distributed average consensus [50])**.** *Let $\boldsymbol{x}^0 \in \mathbb{R}^n$ and $\boldsymbol{\alpha}^0 \in \mathcal{C}(\boldsymbol{L}_-)$. Then $\boldsymbol{u}^k$ converges Q-linearly to $\boldsymbol{u}^* = [\boldsymbol{z}^*; \boldsymbol{\beta}^*]$, with $\boldsymbol{z}^* = \frac{1}{2}\boldsymbol{M}_+\boldsymbol{1}_n\bar{r}$ and $\boldsymbol{\beta}^*$ being the unique vector in $\mathcal{C}(\boldsymbol{M}_-)$ such that $\boldsymbol{M}_-^T\boldsymbol{\beta}^* = \boldsymbol{r} - \boldsymbol{1}_n\bar{r}$, with respect to the $\boldsymbol{G}$-norm:*

$$\|\boldsymbol{u}^{k+1} - \boldsymbol{u}^*\|_{\boldsymbol{G}} \leq \frac{1}{1+\eta}\|\boldsymbol{u}^k - \boldsymbol{u}^*\|_{\boldsymbol{G}}, \tag{2.12}$$

where $\|\boldsymbol{u}^{k+1} - \boldsymbol{u}^*\|_{\boldsymbol{G}} = \sqrt{(\boldsymbol{u}^{k+1} - \boldsymbol{u}^*)^T\boldsymbol{G}\boldsymbol{u}^{k+1} - \boldsymbol{u}^*}$, $\eta = \sqrt{1+\delta} - 1$, $\mu > 1$, and

$$\delta = \min\left\{ \frac{(\mu-1)\lambda_2(\boldsymbol{L}_-)}{\mu\lambda_n(\boldsymbol{L}_+)}, \frac{2\rho\lambda_2(\boldsymbol{L}_-)}{\rho^2\lambda_n(\boldsymbol{L}_+)\lambda_2(\boldsymbol{L}_-) + \mu} \right\}.$$

Furthermore, $\boldsymbol{s}^k$ is R-linearly convergent to $\boldsymbol{s}^*$ as

$$\|\boldsymbol{s}^{k+1} - \boldsymbol{s}^*\|_2 \leq \left(1 + \frac{\sqrt{2\rho\lambda_n(\boldsymbol{L}_-)}}{1+\eta}\right)\|\boldsymbol{u}^k - \boldsymbol{u}^*\|_{\boldsymbol{G}}. \tag{2.13}$$

Indeed, the classical method and gossip based method also converge fast with appropriately

chosen algorithm parameters. These algorithms, however, require the knowledge of an upper bound on r_i's in order to select the right algorithm parameters to deal with the bounded quantization constraint. In Chapter 4, we will show that the local Lagrangian multipliers α_i's in CADMM play a key role in handling the bounded quantization.

2.4 Summary

This chapter introduces CADMM for distributed average consensus and its convergence properties. These results will be used to analyze the proposed quantized CADMM algorithms in the following two chapters.

CHAPTER 3
QUANTIZED CONSENSUS BY THE ADMM: UNBOUNDED QUANTIZATION

This chapter considers distributed average consensus subject to quantized communication constraints. For the present chapter, we do not require the quantizer to be of finite bits. However, if agents' data are bounded and a bound is known, then truncation can be used before quantization to achieve finite-bit data communications.

3.1 Model of Quantized Communication

To model the effect of quantized communications, we assume that each agent can store and compute real values with infinite precision; however, an agent can only transmit quantized data through the channel which are received by its neighbors without any error. The quantization operation is defined as follows. Let $\Delta > 0$ be the quantization resolution. Define the quantization lattice in $\mathbb{R}$ by

$$\Lambda = \{t\Delta : t \in \mathbb{Z}\}.$$

A (uniform) quantizer is a function $\mathcal{Q} : \mathbb{R} \to \Lambda$ that maps a real value to some point in Λ. Among all unbounded quantizers we consider the following two:

- Probabilistic quantizer $\mathcal{Q}_p$ defined as follows: for $y \in [t\Delta, (t+1)\Delta)$,

$$\mathcal{Q}_p(y) = \begin{cases} t\Delta, & \text{with probability } t + 1 - \frac{y}{\Delta}, \\ (t+1)\Delta, & \text{with probability } \frac{y}{\Delta} - t. \end{cases} \tag{3.1}$$

- Rounding quantizer $\mathcal{Q}_d$ which projects $y \in \mathbb{R}$ to its nearest point in Λ:

$$\mathcal{Q}_d(y) = t\Delta, \text{ if } \left(t - \frac{1}{2}\right)\Delta < y \leq \left(t + \frac{1}{2}\right)\Delta. \tag{3.2}$$

We point out that probabilistic quantization is equivalent to a dithered quantization method (see [20, Lemma 2]) while rounding quantization is one of the deterministic quantization schemes. Through the rest of this dissertation, we mean quantizing each of the entries when the quantizer has a vector input. Define $e(y) = \mathcal{Q}(y) - y$ as the quantization error. It is clear that

$$|e_p(y)| \leq \Delta \text{ and } |e_d(y)| \leq \frac{1}{2}\Delta, \text{ for any } y \in \mathbb{R}. \tag{3.3}$$

We will investigate how the quantized communication affects CADMM in the following two sections. We remark that the results of probabilistic and rounding quantizations can easily extend to other dithered and deterministic cases, which will be elaborated in Sections 3.2 and 3.3.

3.2 Probabilistic Quantized CADMM (PQ-CADMM)

For ease of presentation, we only study the probabilistic quantization defined in (3.1). The results can be easily extended to any other dithered quantization as the only information used is the first and second order moments of the probabilistic quantizer output, which are common properties among all dithered quantizers. The properties are stated in the following lemma.

Lemma 3.1 ([51, Lemma 2]). *For every $y \in \mathbb{R}$, it holds that*

$$\mathbb{E}\left[\mathcal{Q}_p(y)\right] = y \ \text{ and } \ \mathbb{E}\left[\left(y - \mathcal{Q}_p(y)\right)^2\right] \leq \frac{\Delta^2}{4}.$$

Furthermore, for any inputs y_1 and y_2, the quantization errors $e_p(y_1)$ and $e_p(y_2)$ are independent and identically distributed (i.i.d.), and are independent of the inputs y_1 and y_2 repsectively.

We use the probabilistic quantization to modify the CADMM update (2.8) as

$$\begin{aligned} x_i^{k+1} &= \frac{1}{1+2\rho|\mathcal{N}_i|}\left(\rho|\mathcal{N}_i|\mathcal{Q}_p\left(x_j^k\right) + \rho\sum_{j\in\mathcal{N}_i}\mathcal{Q}_p\left(x_j^k\right) - \alpha_i^k + r_i\right), \\ \alpha_i^{k+1} &= \alpha_i^k + \rho\left(|\mathcal{N}_i|\mathcal{Q}_p\left(x_i^{k+1}\right) - \sum_{j\in\mathcal{N}_i}\mathcal{Q}_p\left(x_j^{k+1}\right)\right). \end{aligned} \tag{3.4}$$

Notice that x_i^k is also quantized at its own node for the $(k+1)$th update; the reason will be given in Remark 3.5. As illustrated in [17], the above iteration can be interpreted as a stochastic gradient update. Viewed from this point, the quantization operation causes x_i^k to fluctuate around the quantization-free updates (2.8). The convergence claims are given in Theorem 3.1.

Theorem 3.1. *For any $\boldsymbol{x}^0 \in \mathbb{R}^n$ and $\boldsymbol{\alpha}^0 \in \mathcal{C}(\boldsymbol{L}_-)$, the probabilistic quantized CADMM (PQ-CADMM) iteration (3.4) generates $x_i^k, i = 1, 2, \ldots, n$, which converges linearly to the data average $\bar{r}$ in the mean sense as $k \to \infty$, i.e.,*

$$\lim_{k\to\infty}\mathbb{E}\left[x_i^k\right] = \bar{r}, i = 1, 2, \ldots, n.$$

In addition, the variance of x_i^k converges to a finite value which depends on the quantization resolution and the network topology.

Proof. Taking expectation of both sides of (3.4), we have

$$\begin{aligned}\mathbb{E}[x_i^{k+1}] &= \frac{1}{1+2\rho|\mathcal{N}_i|}\left(\rho|\mathcal{N}_i|\mathbb{E}\left[\mathcal{Q}_p(x_i^k)\right] + \rho\sum_{j\in\mathcal{N}_i}\mathbb{E}\left[\mathcal{Q}_p(x_j^k)\right] - \mathbb{E}[\alpha_i^k] + r_i\right),\\ \mathbb{E}[\alpha_i^{k+1}] &= \mathbb{E}[\alpha_i^k] + \rho\left(|\mathcal{N}_i|\mathbb{E}\left[\mathcal{Q}_p(x_i^{k+1})\right] - \sum_{j\in\mathcal{N}_i}\mathbb{E}\left[\mathcal{Q}_p(x_j^{k+1})\right]\right).\end{aligned} \tag{3.5}$$

Noting that Lemma 3.1 implies $\mathbb{E}\left[\mathcal{Q}_p(x_i^k)\right] = \mathbb{E}[x_i^k]$ and $\mathbb{E}\left[\mathcal{Q}_p(x_j^k)\right] = \mathbb{E}[x_j^k]$, we see that (3.5) takes exactly the same iterations in the mean sense as CADMM for distributed average consensus. By initializing $\boldsymbol{\alpha}^0 \in \boldsymbol{L}_-$, we have $\mathbb{E}[\boldsymbol{\alpha}^0] = \boldsymbol{\alpha}_0$ lies in $\mathcal{C}(\boldsymbol{L}_-)$, too. The linear convergence of $\mathbb{E}[x_i^k]$ to $\bar{r}$ is thus ensured due to Theorem 2.2.

Since Lemma 3.1 also indicates the bounded variance of quantization error, the second claim follows directly from [17, Proposition 3]. □

We notice that the convergence of $\mathbb{E}[x_i^k]$ to $\bar{r}$ with bounded variance does not imply that $\boldsymbol{x}^k$ reaches a consensus when $k \to \infty$. Nevertheless, a simple method fixes this problem. The idea is to calculate the running average $\bar{x}_i^k = \frac{1}{k}\sum_{l=1}^k x_i^l, k \geq 1$ at node i. Since the variance is bounded, one can use similar steps as in [17] together with Chebyshev's inequality to get that $\bar{x}_i^k \to \bar{r}$ in probability.

3.3 Deterministic Quantized CADMM (DQ-CADMM)

Deterministic quantization is usually much harder to handle as the quantization error is not stochastic. Unlike probabilistic quantization, the accumulated error term can blow up; there have been a few methods proposed to counter such difficulties (see [12, 22, 24]), yet the resulting algorithms either do not guarantee a consensus or reach a consensus with an error from the desired average that depends on the number of agents, the quantization resolution, and the agents' data. To analyze the consensus reaching result, we first find a finite upper bound on the accumulated error term and then utilize the property and the initialization condition of local Lagrangian multipliers .

Let the local data x_i^k be also quantized for the $(k+1)$-th update at node i. The updates become

$$\begin{aligned} x_i^{k+1} &= \frac{1}{1+2\rho|\mathcal{N}_i|}\left(\rho|\mathcal{N}_i|\mathcal{Q}_d\left(x_j^k\right)+\rho\sum_{j\in\mathcal{N}_i}\mathcal{Q}_d\left(x_j^k\right)-\alpha_i^k+r_i\right), \\ \alpha_i^{k+1} &= \alpha_i^k+\rho\left(|\mathcal{N}_i|\mathcal{Q}_d\left(x_i^{k+1}\right)-\sum_{j\in\mathcal{N}_i}\mathcal{Q}_d\left(x_j^{k+1}\right)\right). \end{aligned} \tag{3.6}$$

We can rewrite $\mathcal{Q}_d(x_i^k)=x_i^k+e_d(x_i^k)$. Then the α_i-update is equivalent to

$$\alpha_i^{k+1}=\alpha_i^k+\rho\left(|\mathcal{N}_i|x_i^{k+1}-\sum_{j\in\mathcal{N}_i}x_j^{k+1}\right)+\rho\left(|\mathcal{N}_i|e_d(x_i^{k+1})-\sum_{j\in\mathcal{N}_i}e_d(x_j^{k+1})\right),$$

or written in the matrix form,

$$\boldsymbol{\alpha}^{k+1}=\boldsymbol{\alpha}^k+\rho\boldsymbol{L}_-\boldsymbol{x}^{k+1}+\rho\boldsymbol{L}_-\boldsymbol{e}_d(\boldsymbol{x}^{k+1}), \tag{3.7}$$

where $\boldsymbol{e}_d(\boldsymbol{x}^{k+1})$ denotes the vector concatenating all $e_d(x_i^{k+1})$. Recalling the CADMM update (2.9), we can write the matrix form of (3.6) as

$$\boldsymbol{s}^{k+1}=\boldsymbol{D}(\boldsymbol{s}^k+\boldsymbol{s}_x^k)+\boldsymbol{s}_\alpha^k \tag{3.8}$$

where $\boldsymbol{s}_x^k=\left[\boldsymbol{e}_d(\boldsymbol{x}^k);\boldsymbol{0};\boldsymbol{0}\right]$ and $\boldsymbol{s}_\alpha^k=\left[\boldsymbol{0};\rho\boldsymbol{L}_-\boldsymbol{e}_d(\boldsymbol{x}^{k+1});\boldsymbol{0}\right]$. It is important to note the update above update is deterministic, i.e., given $\boldsymbol{s}^{k_1}=\boldsymbol{s}^{k_2}$ that are some states, we must have $\boldsymbol{s}^{k_1+1}=\boldsymbol{s}^{k_2+1}$. The result of (3.6) is characterized in the following theorem.

Theorem 3.2. *Consider the deterministic quantized CADMM (DQ-CADMM) iteration (3.6). Let $\boldsymbol{x}^0\in\mathbb{R}^n$ and $\boldsymbol{\alpha}^0\in\mathcal{C}(\boldsymbol{L}_-)$. Then there exists a finite time iteration $k_0\geq 1$ such that for $k\geq k_0$, all the quantized variable values*

- *either converge to the same quantization value:*

$$\mathcal{Q}_d\left(x_1^k\right)=\mathcal{Q}_d\left(x_2^k\right)=\cdots=\mathcal{Q}_d\left(x_n^k\right)\triangleq x_{\mathcal{Q}_d}^*,$$

- *or cycle around the average $\bar{r}$ with a finite period $T \geq 2$, i.e., $\mathcal{Q}_d\left(x_i^k\right) = \mathcal{Q}_d\left(x_i^{k+T}\right), i = 1, 2, \ldots, n$, and*

$$\frac{1}{T}\sum_{l=1}^{T}\mathcal{Q}_d\left(x_1^{k+l}\right) = \frac{1}{T}\sum_{l=1}^{T}\mathcal{Q}_d\left(x_2^{k+l}\right) = \cdots = \frac{1}{T}\sum_{l=1}^{T}\mathcal{Q}_d\left(x_n^{k+l}\right) \triangleq \bar{x}^*_{\mathcal{Q}_d}. \tag{3.9}$$

Furthermore, we have the following error bound for $x^*_{[\mathcal{Q}_d]} \in \left\{x^*_{\mathcal{Q}_d}, \bar{x}^*_{\mathcal{Q}_d}\right\}$:

$$\left|x^*_{[\mathcal{Q}_d]} - \bar{r}\right| \leq \left(1 + 4\rho\frac{m}{n}\right)\frac{\Delta}{2}, \tag{3.10}$$

where the upper bound is tight if DQ-CADMM converges.

Proof. See Section 3.7. □

Remark 3.1. *The result that deterministic quantization may lead the average consensus algorithm to either convergent or cyclic cases is also reported in [24]. Similar to theirs, one can use the history of agents' variables, e.g., running average, to achieve asymptotic convergence at each node. Different to theirs, while their algorithm can make local variable values close to the true average in cyclic cases without guaranteeing a consensus, our algorithm can reach a consensus but does not make the error arbitrarily small in general.*

Remark 3.2. *We shall mention that $x^*_{\mathcal{Q}_d}$ or $\bar{x}^*_{\mathcal{Q}_d}$ need not be unique. This is because, unlike CADMM, $\|\boldsymbol{u}^k - \boldsymbol{u}^*\|_{\boldsymbol{G}}$ in DQ-CADMM need not decrease monotonically due to the quantization that occurs on $\boldsymbol{x}^k$ at each update. Note also that practical consensus value does not necessarily meet the error bound and we usually have smaller errors than (3.10) in practice (see simulations). We hence expect better consensuses when $(\boldsymbol{x}^0, \boldsymbol{\alpha}^0)$ are initialized closer to the ideal optima, which leads to a two-stage algorithm for quantized consensus in Section 3.4.*

Remark 3.3. *An interesting observation of our main result is the ADMM parameter ρ. While a small ρ indicates a small consensus error bound, the current chapter does not quantify how it affects the convergence time k_0 as well as the cyclic period T. We do not study the optimal selection*

of ρ but simply set $\rho = 1$ for the present chapter. We will discuss the effect of ρ in more details in Chapter 4 .

Remark 3.4. *Theorem 3.2 for rounding quantization extends straightforward to other deterministic quantizations as the only information used in our proof is the bounded quantization error. In contrast with [9, 12] where the algorithms may fail for some deterministic quantization schemes, e.g., the rounding quantization, our results work for all deterministic quantization schemes as long as a finite quantization error bound is provided.*

Remark 3.5. *In both PQ-CADMM and DQ-CADMM iterations, x_i^k is quantized for the $(k+1)$th update at node i even though nodes can compute and store real values with infinite precision. The reason is to guarantee that $\boldsymbol{\alpha}^k$ lies in the column space of $\boldsymbol{L}_-$ and thus the ideal CADMM update in either PQ-CADMM or DQ-CADMM (cf. Eq. (3.8)) possesses the linear convergence property given in Theorem 2.2. If we do not quantize x_i^k at its own node, Theorem 3.1 still holds due to $\mathbb{E}[\mathcal{Q}_p(x_i^k)] = \mathbb{E}[x_i^k]$ while Theorem 3.2 may fail.*

3.4 A Two-Stage Algorithm for Quantized Consensus

Let us summarize the two quantized versions of CADMM: PQ-CADMM converges linearly to the data average in the mean sense, but it does not guarantee a consensus within finite iterations; DQ-CADMM, on the other hand, either converges to a consensus or cycles with the same mean of quantized variable values over one period at each node after a finite-time iteration, but results in an error from the true average.

As discussed in Remark 3.2, we can first run PQ-CADMM $2K$ times to obtain $\bar{x}_i = \frac{1}{K}\sum_{k=K+1}^{2K} x_i^k$, which is a reasonable estimate of $\bar{r}$ at node i according to Theorem 3.1. Here K can be chosen such that $\mathbb{E}[x_i^K]$ is close enough to $\bar{r}$ when we have the knowledge of agents' data and the network topology. Otherwise, we can simply pick $K = \left\lceil 10n\left(\log_{10}(\frac{1}{\Delta}+1)+1\right)\max\{-\log_{10}\rho, 1\}\right\rceil$ which works well in practice, or as large as permitted. Also, $\bar{\alpha}_i = \frac{1}{K}\sum_{k=K+1}^{2K}\alpha_i^k$ is a good estimate of $\alpha_i^* = r_i - \bar{r}$, and that $\bar{\boldsymbol{\alpha}} = [\bar{\alpha}_1; \bar{\alpha}_2; \cdots; \bar{\alpha}_n] = \frac{1}{K}\sum_{k=K+1}^{2K}\boldsymbol{\alpha}^k$ satisfies the initialization

condition as $\boldsymbol{\alpha}^k$ lies in the column space of $\boldsymbol{L}_-$. We can therefore run DQ-CADMM with this $\bar{x}_i$ and $\bar{\alpha}_i$ as initial values. The probabilistic quantized CADMM followed by deterministic quantized CADMM (PQDQ-CADMM) is presented in Algorithm 3.1.

Algorithm 3.1 PQDQ-CADMM for quantized consensus

Require: Initialize $\rho > 0$, $K = \left\lceil 10n\left(\log_{10}(\frac{1}{\Delta}+1)+1\right)\max\{-\log_{10}\rho, 1\}\right\rceil$, $\boldsymbol{x}^0 = \boldsymbol{0}$, and $\boldsymbol{\alpha}^0 = \boldsymbol{0}$.

1: **for** $k = 0, 1, \cdots, 2K-1$, every node i **do**

$$x_i^{k+1} \leftarrow \frac{1}{1+2\rho|\mathcal{N}_i|}\left(\rho|\mathcal{N}_i|\mathcal{Q}_p\left(x_i^k\right) + \rho\sum_{j\in\mathcal{N}_i}\mathcal{Q}_p\left(x_j^k\right) - \alpha_i^k + r_i\right),$$

$$\alpha_i^{k+1} \leftarrow \alpha_i^k + \rho\left(|\mathcal{N}_i|\mathcal{Q}_p\left(x_i^{k+1}\right) - \sum_{j\in\mathcal{N}_i}\mathcal{Q}_p\left(x_j^{k+1}\right)\right).$$

2: **end for**

3: $x_i^{2K} \leftarrow \frac{1}{K}\sum_{l=K+1}^{2K} x_i^l$, $\alpha_i^{2K} \leftarrow \frac{1}{K}\sum_{l=K+1}^{2K}\alpha_i^l$, and $k \leftarrow 2K$.

4: **repeat**

5: every node i **do**

$$x_i^{k+1} \leftarrow \frac{1}{1+2\rho|\mathcal{N}_i|}\left(\rho|\mathcal{N}_i|\mathcal{Q}_d\left(x_i^k\right) + \rho\sum_{j\in\mathcal{N}_i}\mathcal{Q}_d\left(x_j^k\right) - \alpha_i^k + r_i\right),$$

$$\alpha_i^{k+1} \leftarrow \alpha_i^k + \rho\left(|\mathcal{N}_i|\mathcal{Q}_d\left(x_i^{k+1}\right) - \sum_{j\in\mathcal{N}_i}\mathcal{Q}_d\left(x_j^{k+1}\right)\right).$$

6: $k \leftarrow k+1$

7: **until** a predefined stopping criterion (e.g., a maximum iteration number) is satisfied.

3.5 Simulations

This section investigates the performance of DQ-CADMM and PQDQ-CADMM via numerical examples. To construct a connected graph with n nodes and m edges, we first generate a complete graph consisting of n nodes, and then randomly remove $\frac{n(n-1)}{2} - m$ edges while ensuring that the

network stays connected. Set $\Delta = 1$ throughout this section and assume that agents' data have very high variances in large networks, e.g., let $r_i \sim \mathcal{N}(0, n^4)$. Throughout the rest of this dissertation, the average simulated result is taken with respect to both graph and agents' data; that is, both graph and agents' data are randomly generated at each run.

3.5.1 Performance Comparison

We compare our algorithms with those that use deterministic quantization to reach a consensus, i.e., the gossip based method in [22] and the classical method in [12]. The simulation settings are

- PQDQ-CADMM: Set $\rho = 1$.
- DQ-CADMM: Set $\rho = 1$, $\boldsymbol{x}^0 = \mathbf{0}$ and $\boldsymbol{\alpha}^0 = \mathbf{0}$.
- Gossip based method: We randomly pick one edge in $\mathcal{A}$ and perform the updating, i.e., if $(i, j) \in \mathcal{A}$ is chosen, then $x_i^{k+1} = x_j^{k+1} = \frac{1}{2}\left(\mathcal{Q}_d(x_i^k) + \mathcal{Q}_d(x_j^k)\right)$.
- Classical method: Let $\boldsymbol{W}$ denote the weight matrix of the graph $\mathcal{G}_d = \{\mathcal{V}, \mathcal{A}\}$. The updating rule is then given by $\boldsymbol{x}^{k+1} = \boldsymbol{W}\mathcal{Q}_{rd}(\boldsymbol{x}^k)$ where the subscript $\mathcal{Q}_{rd}(\cdot)$ denotes the rounding down quantization. We utilize the Metropolis weights defined in [52]:

$$W_{ij} = \begin{cases} (1 + \max\{|\mathcal{N}_i|, |\mathcal{N}_j|\})^{-1}, & (i,j) \in \mathcal{A}, \\ 1 - \sum_{k \in \mathcal{N}_i} W_{ik}, & i = j, \\ 0, & \text{otherwise.} \end{cases}$$

We simulate a connected network with $n = 50$ nodes and $m = 500$ edges. Define the iterative error as $\|\mathcal{Q}(\boldsymbol{x}^k) - \mathbf{1}_n\bar{r}\|_2/\sqrt{n}$ where $\mathcal{Q}(\cdot)$ denotes the quantization scheme in the respective algorithms. Plotted in Fig. 3.1 is the iterative error with each value being the average of 1000 runs (the iterative error is $\|\bar{\boldsymbol{x}}^k - \mathbf{1}_n\bar{r}\|_2$ for the PQ-CADMM part of PQDQ-CADMM). Note that we start the plot of PQDQ-CADMM from the $(K+1)$-th iteration as its first K iterations are used only

to reach a neighborhood of $\bar{r}$; at the $(2K+1)$-th iteration, $\mathcal{Q}_d\left(\boldsymbol{x}^{2K+1}\right)$ is updated based on the running average of the $(K+1)$-th iteration to the $2K$-th iteration. The figure indicates that all the four algorithms converge to a consensus at one of the quantization levels. The average consensus error of DQ-CADMM is 1.21, which is much smaller than the upper bound $(\frac{1}{2}+\frac{2m}{n})\Delta = 20.5$. One can also see that PQDQ-CADMM converges almost immediately after the $2K$-th iteration.

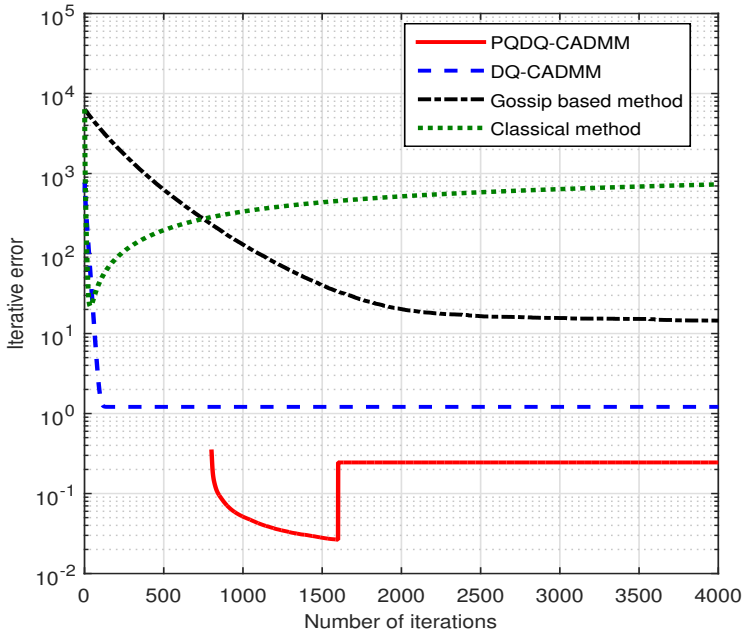


Fig. 3.1: Iterative error versus iterations where each plotted value is the average of 1000 runs.

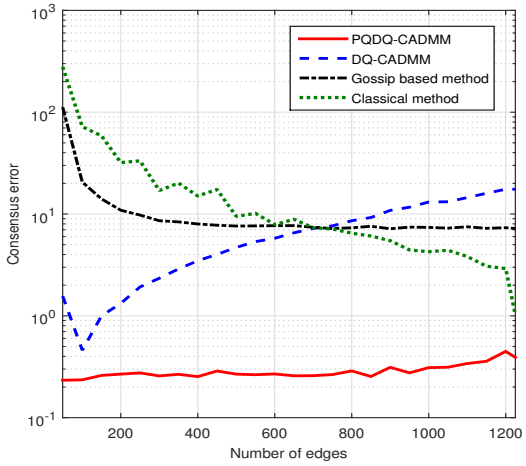
Consensus error: In Fig. 3.2a we fix $n = 50$ and vary m until the graph is complete. The gossip based method and the classical method have decreasing consensus errors as m increases. The consensus error of DQ-CADMM, however, becomes larger as the average degree and therefore the error bound increase. PQDQ-CADMM has the smallest consensus error whose average of 100 runs is less than 0.40 for all m. We then fix $m = 400$ and let n vary. Fig. 3.2b shows that the gossip based method and the classical method have increasing consensus errors as n increases. The consensus error of DQ-CADMM, on the contrary, decreases when n becomes larger. PQDQ-CADMM also has the smallest consensus error in this case. In the last setting we fix the average degree $\frac{2m}{n} = 10$ while varying n. The classical method and the gossip based method then both

have increasing consensus errors when n and thus the range of agents' data increase. The consensus error of DQ-CADMM is relatively small compared with the upper bound $(0.5 + \frac{2m}{n})\Delta = 10.5$ and decreases when n becomes larger. The proposed PQDQ-CADMM algorithm still has the smallest consensus error whose average of 100 runs is less than 0.2 for all n.

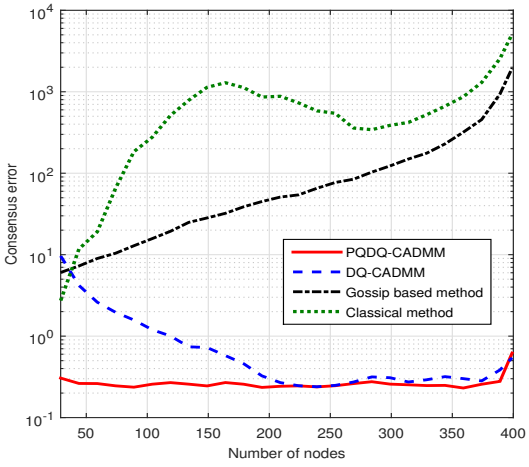
We conclude that the consensus error of the gossip based method and the classical method depends on the average degree of the graph as well as the range of agents' data. Note that their consensus errors can be extremely large for a sparsely connected graph. DQ-CADMM has an increasing consensus error when the average degree increases while PQDQ-CADMM performs almost the same for all network structures in terms of the consensus error.

Convergence time: We study the convergence time of the four algorithms via numerical examples in Fig. 3.3. Since the gossip based method involves only one edge and the other three methods utilize all the edges at each iteration, we plot also the quotient of the convergence time of the gossip based method divided by the number of edges, namely, Gossip based method adjusted, in the figure.

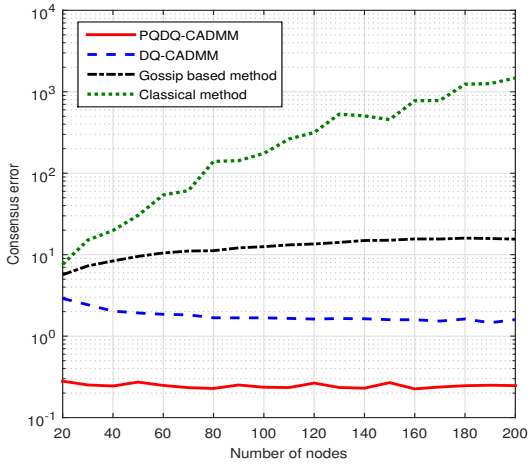
In Fig. 3.3a, the gossip based method and the classical method converge slower as the graph becomes sparser. When the average degree is fixed, they have longer convergence time as n increases. Therefore, the convergence time of the gossip based method and the classical method is also affected by the average degree of the graph and the range of agents' data. Different from the gossip based and classical methods, we see in Fig. 3.3a that the convergence time of DQ-CADMM increase as the graph becomes denser. In Fig. 3.3b and Fig. 3.3c, however, the convergence time also increases while the graph becomes sparser, which is possibly because of the increased distance between starting variable values and optimal variable values. For PQDQ-CADMM, we observe that the significant portion of its convergence time is spent on achieving an approximate estimate of $\bar{r}$, i.e., running PQ-CADMM with $2K$ iterations. With good starting points, DQ-CADMM converges almost immediately.



(a)

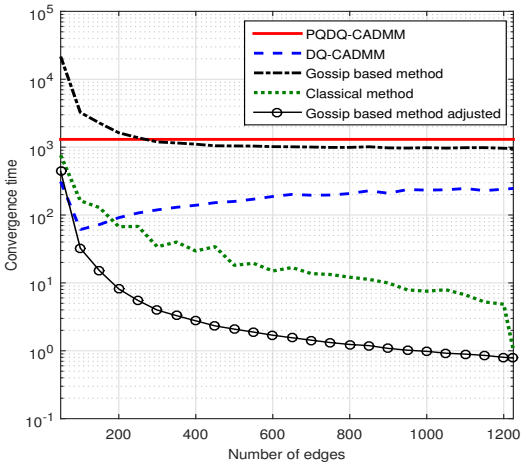


(b)

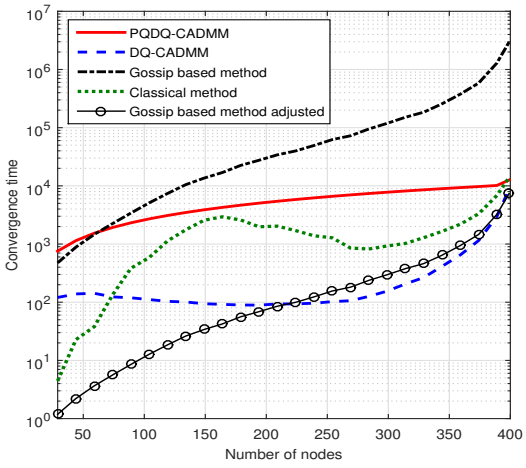


(c)

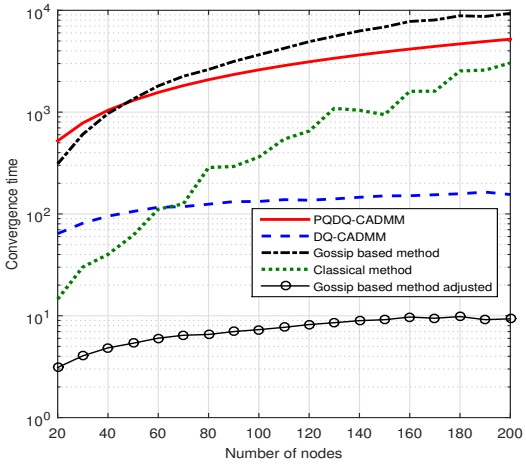
Fig. 3.2: Consensus error of the four algorithms where $\Delta = 1$ and the plotted values are the average of 100 runs; (a) fixing $n = 50$ and varying $m \in [49, 1225]$, (b) fixing $m = 400$ and varying $n \in [29, 399]$, (c) fixing $\frac{2m}{n} = 10$ and varying $n \in [20, 200]$.



(a)



(b)



(c)

Fig. 3.3: Convergence time of the four algorithms where $\Delta = 1$ and the plotted values are the average of 100 runs; (a) $n = 50$ and $m \in [49, 1225]$, (b) $m = 400$ and $n \in [29, 399]$, (c) $\frac{2m}{n} = 10$ and $n \in [20, 200]$.

3.5.2 Different Quantization Resolutions

We next consider the effect of the quantization resolution on PQDQ-CADMM. Fig. 3.4 plots consensus errors of PQDQ-CADMM with $n = 50$ and $m \in [49, 1225]$ for $\Delta \in \{0.02, 0.1, 0.5, 2.5\}$. The consensus error tends to increase on the average as the quantization resolution becomes larger, which is not surprising since a coarse quantization indicates a higher loss of information at each update. We then calculate the ratio of the consensus error to the quantization resolution: the plotted values, which are the averages of 100 runs, all lie in $(0.227\Delta, 0.337\Delta)$ and the variances are less than 0.051. Moreover, the convergence time of each quantization resolution has a mean of $(2K + 2.1)$ iterations and a variance less than 0.0008, which coincides with our previous analysis that PQDQ-CADMM converges immediately after the first $2K$ iterations.

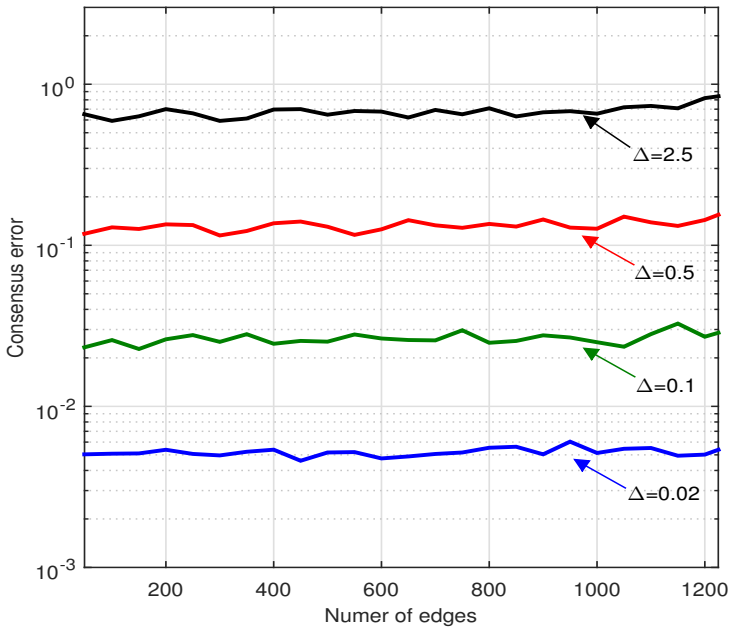


Fig. 3.4: Consensus error of PQDQ-CADMM with different quantization resolutions, i.e., $\Delta \in \{0.02, 0.1, 0.5, 2.5\}$, for $n = 50$ and $m \in [49, 1225]$; each plotted value is the average of 100 runs.

3.5.3 Cyclic Case

While we prove that DQ-CADMM either converges or cycles in Theorem 4, we note the above numerical examples all lead to reach convergence results. Indeed, the proposed deterministic algorithms, DQ-CADMM and PQDQ-CADMM, converges in most cases as shown by the following simulation. For connected networks with n nodes, we consider star graph which has the smallest average degree, randomly generated graph that has intermediate average degree, and complete graph that has the largest average degree. The result is given in Fig. 3.5 where the y-axis represents the number of cyclic cases in 10^4 trials. Clearly, DQ-CADMM and PQDQ-CADMM with fixed parameter $\rho = 1$ converge in most cases, particularly with large networks. We will study the cyclic case in more details in Chapter 4.

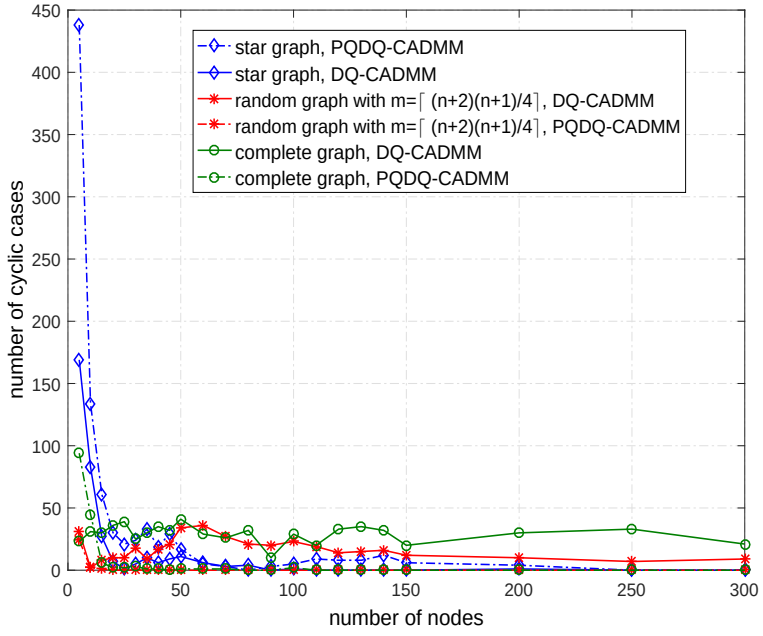


Fig. 3.5: Number of cyclic cases in 10^4 trials.

3.6 Summary

In this chapter, we propose two quantized versions of CADMM: PQ-CADMM and DQ-CADMM. PQ-CADMM converges linearly to the data average in the mean sense, but it does not guarantee a consensus within finite iterations. DQ-CADMM, on the other hand, either converges to a consensus or cycles with the same mean of quantized variable values over one period at each node after a finite-time iteration but results in an error from the true average. While deterministic quantization is somewhat unfavorable due to the cyclic behavior, a notable fact is that there is no randomness involved. We believe that this property will be useful to some applications, e.g., consensus based detection using one-bit communications as discussed in Chapter 5.

3.7 Proof of Theorem 3.2

Proof. We prove that DQ-CADMM either converges or cycles after a finite-time iteration and then use this fact to derive the error bound.

We see from (3.7) that $\boldsymbol{\alpha}^k$ must lie in the column space of $\boldsymbol{L}_-$ if $\boldsymbol{\alpha}^0$ is initialized in the column space of $\boldsymbol{L}_-$. Following (3.8), we have

$$\begin{aligned}
\boldsymbol{s}^k &= \boldsymbol{D}(\boldsymbol{s}^{k-1} + \boldsymbol{s}_x^{k-1}) + \boldsymbol{s}_\alpha^{k-1} \\
&= \boldsymbol{D}\left(\boldsymbol{D}(\boldsymbol{s}^{k-2} + \boldsymbol{s}_x^{k-2}) + \boldsymbol{s}_\alpha^{k-2}\right) + \boldsymbol{D}\boldsymbol{s}_x^{k-1} + \boldsymbol{s}_\alpha^{k-1} \\
&= \cdots \\
&= \boldsymbol{D}^k \boldsymbol{s}^0 + \left(\sum_{i=1}^{k} \boldsymbol{D}^i \boldsymbol{s}_x^{k-i} + \sum_{j=0}^{k-1} \boldsymbol{D}^j \boldsymbol{s}_\alpha^{k-1-j} \right). \qquad (3.11)
\end{aligned}$$

The first term is simply the ideal CADMM update which converges to a finite value. We will show that the accumulated error term $\sum_{i=1}^{k} \boldsymbol{D}^i \boldsymbol{s}_x^{k-i} + \sum_{j=0}^{k-1} \boldsymbol{D}^j \boldsymbol{s}_\alpha^{k-1-j}$ is bounded and hence that $\boldsymbol{s}^k$ is bounded. Notice that $\boldsymbol{D}^i \boldsymbol{s}_x^{k-i}$ is the i-th update of CADMM with the initial value $\boldsymbol{s}_x^{k-i}$. Let $\boldsymbol{u}_{k-i}^l = [\boldsymbol{z}_{k-i}^l; \boldsymbol{\beta}_{k-i}^l]$ be the vector that concatenates the primal and dual variables in the ADMM

iteration (2.6), with initial values $\boldsymbol{z}_{k-i}^0 = \frac{1}{2}\boldsymbol{M}_+^T \boldsymbol{e}_d^{k-i}$ and $\boldsymbol{\beta}_{k-i}^0 = \mathbf{0}$ corresponding to $\boldsymbol{s}_x^k = [\boldsymbol{e}_d^k; \mathbf{0}; \mathbf{0}]$. With $\boldsymbol{G}$ defined in (2.11), we obtain

$$\begin{aligned}\|\boldsymbol{u}_{k-i}^0\|_{\boldsymbol{G}}^2 &= \rho \left\| \frac{1}{2}\boldsymbol{M}_+^T \boldsymbol{e}_d^{k-i} \right\|_2^2 \\ &\leq \frac{1}{4}\rho 2\lambda_n(\boldsymbol{L}_+)\|\boldsymbol{e}_d^{k-i}\|_2^2 \\ &\leq \frac{1}{8}\rho n \Delta^2 \lambda_n(\boldsymbol{L}_+)(\boldsymbol{M}_+),\end{aligned}$$

where the last inequality is from (3.3). Since Theorem 2.1 indicates the form of $\boldsymbol{D}^*$, we get $\boldsymbol{D}^* \boldsymbol{s}_x^{k-i} = \mathbf{0}$, i.e., $\boldsymbol{x}_{k-i}^* = \mathbf{0}$ and $\boldsymbol{\alpha}_{k-i}^* = \mathbf{0}$. Therefore, $\boldsymbol{u}_{k-i}^* = [\boldsymbol{z}_{k-i}^*; \boldsymbol{\beta}_{k-i}^*] = \mathbf{0}$ from Lemma 2.1 and the fact that $\boldsymbol{z}_{k-i}^* = \frac{1}{2}\boldsymbol{M}_+^T \boldsymbol{x}_{k-i}^*$. Noting also that the initialization $\boldsymbol{z}_{k-i}^0$ and $\boldsymbol{\beta}_{k-i}^0$ meet the condition of Theorem 2.2, we thus have

$$\begin{aligned}\|\boldsymbol{D}^i \boldsymbol{s}_x^{k-i}\|_2 &= \|(\boldsymbol{D}^i - \boldsymbol{D}^*)\boldsymbol{s}_x^{k-i}\|_2 \\ &\overset{(a)}{\leq} \left(1 + \sqrt{\frac{\rho}{1+\delta} 2\lambda_n(\boldsymbol{L}_-)}\right) \|\boldsymbol{u}_{k-i}^{i-1} - \boldsymbol{u}_{k-i}^*\|_{\boldsymbol{G}} \\ &\overset{(b)}{\leq} \frac{1}{4}\left(1 + \sqrt{\frac{\rho}{1+\delta} 2\lambda_n(\boldsymbol{L}_-)}\right)\left(\sqrt{\frac{1}{1+\delta}}\right)^{i-1} \sqrt{2\rho n \lambda_n(\boldsymbol{L}_+)},\end{aligned} \tag{3.12}$$

where (a) and (b) are due to Theorem 2.2 together with the fact that $\boldsymbol{u}_{k-i}^* = \mathbf{0}$. Similarly, we have for $j \geq 1$,

$$\|\boldsymbol{D}^j \boldsymbol{s}_\alpha^{k-1-j}\|_2 \leq \frac{1}{4}\left(1 + \sqrt{\frac{\rho}{1+\delta} 2\lambda_n(\boldsymbol{L}_-)}\right)\left(\sqrt{\frac{1}{1+\delta}}\right)^{j-1} \Delta \sigma_{\max}(\boldsymbol{M}_-)\sqrt{\rho n}, \tag{3.13}$$

and when $j = 0$,

$$\|\boldsymbol{D}^j \boldsymbol{s}_\alpha^{k-1-j}\|_2 = \|\boldsymbol{s}_\alpha^{k-1}\|_2 \leq \frac{1}{4}\rho \Delta \sqrt{2n\lambda(\boldsymbol{L}_-)}. \tag{3.14}$$

Therefore,

$$\begin{aligned}
&\left\| \sum_{i=1}^{k} \boldsymbol{D}^i \boldsymbol{s}_x^{k-i} + \sum_{j=0}^{k-1} \boldsymbol{D}^j \boldsymbol{s}_\alpha^{k-1-j} \right\|_2 \\
&\leq \sum_{i=1}^{k} \|\boldsymbol{D}^i \boldsymbol{s}_e^{k-i}\|_2 + \sum_{j=0}^{k-1} \|\boldsymbol{D}^j \boldsymbol{s}_\alpha^{k-1-j}\|_2 \\
&\leq \|\boldsymbol{s}_\alpha^{k-1}\|_2 + \sum_{i=1}^{k} \left(\|\boldsymbol{D}^i \boldsymbol{s}_e^{k-i}\|_2 + \|\boldsymbol{D}^i \boldsymbol{s}_\alpha^{k-1-i}\|_2 \right) \\
&\overset{(a)}{\leq} \frac{1}{4}\rho\Delta\sqrt{2n\lambda(\boldsymbol{L}_-)} + \left(1 + \sqrt{\frac{\rho}{1+\delta}2\lambda(\boldsymbol{L}_-)}\right) \\
&\quad \times \frac{1}{4}\Delta\sqrt{\rho N}\left(\sqrt{2\lambda(\boldsymbol{L}_-)} + \sqrt{2\lambda(\boldsymbol{L}_+)}\right) \sum_{i=1}^{k} \left(\sqrt{\frac{1}{1+\delta}}\right)^{i-1}
\end{aligned} \tag{3.15}$$

where (a) is from (3.12)-(3.14). Then (3.15) must be finite for $k = 1, 2, \ldots,$ as $\delta > 0$, and thus $\boldsymbol{s}^k$ is bounded. An important fact from (3.8) is that the update of $\boldsymbol{s}^{k+1}$ and hence $\boldsymbol{s}_x^{k+1}$ is fully determined by $\boldsymbol{s}^k + \boldsymbol{s}_x^k$ due to the deterministic quantization and the CADMM update. Recalling that $\|\boldsymbol{s}_x^k\|_2 = \|\boldsymbol{e}_d^k\|_2 \leq \frac{\Delta}{2}\sqrt{n}$ and that $\boldsymbol{s}^k + \boldsymbol{s}_x^k = [\mathcal{Q}_d(\boldsymbol{x}^k); \boldsymbol{\alpha}^k; \boldsymbol{r}]$ with each entry of $\mathcal{Q}_d(\boldsymbol{x}^k)$ being a multiple of Δ, each entry of $\boldsymbol{\alpha}$ being a multiple of $\rho\Delta$, and $\boldsymbol{r}$ being fixed, we conclude that there are only finite possible states of $\boldsymbol{s}^k + \boldsymbol{s}_x^k$. Therefore, $\boldsymbol{s}^k$ is either convergent or cyclic with a finite period $T \geq 2$ after a finite-time iteration.

We next consider error bounds for the consensus value. The consensus error may be studied directly by calculating the accumulated error term in (3.11). However, the bound in (3.15) is quite loose in general since it results from the worst case. We alternatively derive the error bounds in the respective case using the fact that DQ-CADMM either converges or cycles.

Convergent case: The convergence of DQ-CADMM implies that $\boldsymbol{s}^{k+1} = \boldsymbol{s}^k$ for $k \geq k_0$, and hence

$$\boldsymbol{0} = \boldsymbol{\alpha}^{k+1} - \boldsymbol{\alpha}^k = \rho \boldsymbol{L}_- \mathcal{Q}_d(\boldsymbol{x}^{k+1}).$$

Since $\boldsymbol{L}_-$ is the Laplacian matrix of a connected graph $\mathcal{G}_u$, we must have that $\mathcal{Q}_d(\boldsymbol{x}^{k+1})$ reaches a consensus. Now let $x_{\mathcal{Q}_d}^* \in \Lambda$ denote the convergent quantized value. Then $\mathcal{Q}_d(x_i^\infty) = x_{\mathcal{Q}_d}^*$ for

$i = 1, 2, \ldots, n$, and $x_i^\infty = x_{\mathcal{Q}_d}^* - e_i^*$. Summing up both sides of (3.6) from $i = 1$ to n, we have

$$\sum_{i=1}^{n}(1 + 2\rho|\mathcal{N}_i|)\left(x_{\mathcal{Q}_d}^* - e_i^*\right) = \sum_{i=1}^{n}\left(\rho|\mathcal{N}_i|x_{\mathcal{Q}_d}^* + \rho\sum_{j\in\mathcal{N}_i} x_{\mathcal{Q}_d}^* + r_i\right),$$

which is equivalent to

$$x_{\mathcal{Q}_d}^* = \frac{1}{n}\sum_{i=1}^{n} r_i + \frac{1}{n}\sum_{i=1}^{n}(1 + 2\rho|\mathcal{N}_i|)e_i^*.$$

Here we use the fact that $\boldsymbol{\alpha}^k$ lies in the column space of $\boldsymbol{L}_-$, i.e., $\boldsymbol{\alpha}^k = \boldsymbol{L}_-\boldsymbol{b}^k$ where $\boldsymbol{b}^k \in \mathbb{R}^n$. Then $\sum_{i=1}^n \alpha_i^k = (\boldsymbol{L}_-\boldsymbol{b}^k)^T\mathbf{1} = (\boldsymbol{b}^k)^T(\boldsymbol{L}_-^T\mathbf{1}) = 0$. Recalling that $\bar{r} = \frac{1}{n}\sum_{i=1}^n r_i$ and $|e_i^*| \le \frac{\Delta}{2}$, we finally obtain

$$\left|x_{\mathcal{Q}_d}^* - \bar{r}\right| \le \left(\frac{1}{2} + \rho\frac{2m}{n}\right)\Delta.$$

The following example shows the tightness of this bound in this convergent case. Consider a simple two-node network with $r_1 = -\frac{3}{2}$ and $r_2 = -\frac{7}{2}$. Set both Δ and ρ to be 1. In this case, we have $m = 1$, $n = 2$, and

$$\boldsymbol{L}_- = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}.$$

We start with $\mathcal{Q}_d(x_1^0) = \mathcal{Q}_d(x_2^0) = -1$ and $\alpha_1^0 = -\alpha_2^0 = 1$. One can easily check $\mathcal{Q}_d(x_1^k) = \mathcal{Q}_d(x_1^k) = -1$ and $\alpha_1^k = -\alpha_2^k = 1, k = 1, 2, \ldots$, in the updates of (3.6). Hence $x_{\mathcal{Q}_d}^* = -1$ and the consensus error is

$$\left|x_{\mathcal{Q}_d}^* - \bar{r}\right| = \frac{3}{2} = \left(\frac{1}{2} + \rho\frac{2m}{n}\right)\Delta.$$

This coincides with the error bound in (3.10).

Cyclic case: When DQ-CADMM cycles with a period T, we must have $\boldsymbol{s}^{k+T} = \boldsymbol{s}^k$. Thus, for $k \ge k_0$, we have that

$$\mathbf{0} = \boldsymbol{\alpha}^{k+T} - \boldsymbol{\alpha}^k = \rho\boldsymbol{L}_-\sum_{l=1}^{T}\mathcal{Q}_d(\boldsymbol{x}^{k+l}),$$

and consequently, $\sum_{l=1}^{T} \mathcal{Q}_d(\boldsymbol{x}_i^{k+l})$ reaches a consensus, i.e., (4.7) is true. Now denote

$$\bar{x}^*_{\mathcal{Q}_d} = \frac{1}{T}\sum_{l=1}^{T} \mathcal{Q}_d(x_i^{k+l}), i = 1, \ldots, n.$$

We then get

$$\left| \bar{x}^*_{\mathcal{Q}_d} - \frac{1}{T}\sum_{l=1}^{T} x_i^{k+l} \right| \leq \frac{1}{T}\sum_{l=1}^{T} \left| \mathcal{Q}_d(x_i^{k+l}) - x_i^{k+l} \right| \leq \frac{\Delta}{2}. \tag{3.16}$$

Summing both sides of (3.6) over one period and dividing the sum by T, we have

$$\frac{1}{T}\sum_{l=1}^{T} x_i^{k+l} = \frac{1}{1+2\rho|\mathcal{N}_i|}\left(2\rho|\mathcal{N}_i|\bar{x}^*_{\mathcal{Q}_d} - \frac{1}{T}\sum_{l=1}^{T} \alpha_i^{k+l} + r_i \right).$$

Finally, using (3.16) and following the same steps as in the convergent case, we conclude that

$$\left| \bar{x}^*_{\mathcal{Q}_d} - \bar{r} \right| \leq \left(\frac{1}{2} + \rho\frac{2m}{n} \right) \Delta.$$

□

CHAPTER 4
QUANTIZED CONSENSUS BY THE ADMM: FINITE-BIT BOUNDED QUANTIZATION

It is noted that both PQ-CADMM and DQ-CADMM in Chapter 3 require infinite bits as the output consists of infinite quantization levels. If it is known *a priori* that agents' data r_i's are bounded and that a bound is known, then existing quantized consensus algorithms can use either truncation or carefully chosen updating weights such that finite-bit quantizers achieve certain guaranteed consensus results. The goal of this chapter is to develop quantized consensus algorithms using finite-bit bounded quantizer for possibly unbounded data.

4.1 Model of Bounded Quantized Communication

Let $\mathcal{X}$ be a nonempty compact convex set in $\mathbb{R}$. We assume without loss of generality that $\mathcal{X} = [-L, L]$ for some $0 < L < \infty$ since we can always translate the set. Further assume that L is a multiple of Δ and set $\Lambda_{\mathcal{X}} = [-L, L] \cap \Lambda$. Define $\mathcal{T}_{\mathcal{X}}(\cdot)$ as the projection operator that maps a real value to the nearest point in $\mathcal{X}$. We define a bounded quantizer $\mathcal{Q}_b : \mathbb{R} \to \Lambda_{\mathcal{X}}$ by first projecting its argument onto $\mathcal{X}$ and then applying the uniform rounding quantizer $\mathcal{Q}_d(\cdot)$ defined in (3.2) to the

projected value, i.e.,

$$\mathcal{Q}_b(\cdot) = \mathcal{Q}_d \circ \mathcal{T}_{\mathcal{X}}(\cdot). \tag{4.1}$$

It is straightforward to see that $|\mathcal{Q}_b(y)| \leq L$ for any $y \in \mathbb{R}$. Therefore, the bounded quantizer $\mathcal{Q}_b(\cdot)$ has $2\frac{L}{\Delta} + 1$ quantization levels which can be represented by $\lceil \log_2(2\frac{L}{\Delta} + 1) \rceil$ bits. We will use this bounded quantizer to modify the CADMM in the next section.

4.2 Bounded Quantized CADMM (BQ-CADMM)

We now modify the CADMM update in (2.8) using the bounded quantizer $\mathcal{Q}_b(\cdot)$, referred to as bounded quantized CADMM (BQ-CADMM), given in Algorithm 4.2.

Algorithm 4.2 BQ-CADMM for quantized consensus

Require: Initialize $x_i^0 = 0$ and $\alpha_i^0 = 0$ for each agent $i, i = 1, 2, \cdots, n$. Set $\rho > 0$ and $k = 0$.

1: **repeat**

2: every node i **do**

$$x_i^{k+1} \leftarrow \frac{1}{1 + 2\rho|\mathcal{N}_i|} \left(\rho|\mathcal{N}_i| \mathcal{Q}_b\left(x_i^k\right) + \rho \sum_{j \in \mathcal{N}_i} \mathcal{Q}_b\left(x_j^k\right) - \alpha_i^k + r_i \right),$$

$$\alpha_i^{k+1} \leftarrow \alpha_i^k + \rho \left(|\mathcal{N}_i| \mathcal{Q}_b\left(x_i^{k+1}\right) - \sum_{j \in \mathcal{N}_i} \mathcal{Q}_b\left(x_j^{k+1}\right) \right).$$

3: $k \leftarrow k + 1$

4: **until** a predefined stopping criterion (e.g., a maximum iteration number) is satisfied.

From Algorithm 4.2, BQ-CADMM is quite similar to PQ-CADMM and DQ-CADMM with the only difference lying in the use of $\mathcal{Q}_b(\cdot)$ on x_i^k. Letting $e_b(y) = \mathcal{Q}_b(y) - y$ denote the quantization error of $\mathcal{Q}_b(\cdot)$, we know that it can be unbounded with unbounded y. As such, it is not clear now how the bounded quantization affects CADMM and how accurate the consensus can be if a

consensus can be reached. Meanwhile, if L is chosen to be large enough such that $\mathcal{T}_{\mathcal{X}}\left(x_i^k\right) = x_i^k$ for all k, then $\mathcal{Q}_b\left(x_i^k\right) = \mathcal{Q}_d\left(x_i^k\right)$ and BQ-CADMM becomes the same as DQ-CADMM. In this sense, we regard DQ-CADMM as a special case of BQ-CADMM. The rest of this section is devoted to investigating the performance of BQ-CADMM.

Noticing first that $\mathcal{Q}_b(\cdot) = \mathcal{Q}_d \circ \mathcal{T}_{\mathcal{X}}(\cdot)$, we can obtain BQ-CADMM by applying the rounding quantizer $\mathcal{Q}_d(\cdot)$ to $\mathcal{T}_{\mathcal{X}}\left(x_i^k\right)$ in the following update:

$$x_i^{k+1} = \frac{1}{1+2\rho|\mathcal{N}_i|}\left(\rho|\mathcal{N}_i|\mathcal{T}_{\mathcal{X}}\left(x_i^k\right) + \rho\sum_{j\in\mathcal{N}_i}\mathcal{T}_{\mathcal{X}}\left(x_j^k\right) - \alpha_i^k + r_i\right), \tag{4.2}$$

$$\alpha_i^{k+1} = \alpha_i^k + \rho\left(|\mathcal{N}_i|\mathcal{T}_{\mathcal{X}}\left(x_i^{k+1}\right) - \sum_{j\in\mathcal{N}_i}\mathcal{T}_{\mathcal{X}}\left(x_j^{k+1}\right)\right). \tag{4.3}$$

Recalling that $\mathcal{T}_{\mathcal{X}}\left(x_i^{k+1}\right)$ is the projection of x_i^{k+1} onto $\mathcal{X}$, we can also write (4.2) as

$$\begin{aligned}\mathcal{T}_{\mathcal{X}}\left(x_i^{k+1}\right) &= \arg\min_{\tilde{x}\in\mathcal{X}} \frac{1}{2}(\tilde{x}-r_i)^2 + \rho|\mathcal{N}_i|\tilde{x}^2 - \left(\rho|\mathcal{N}_i|\mathcal{T}_{\mathcal{X}}\left(x_i^k\right) + \rho\sum_{j\in\mathcal{N}_i}\mathcal{T}_{\mathcal{X}}\left(x_j^k\right) - \alpha_i^k\right)\tilde{x} \\ &= \arg\min_{\tilde{x}} \frac{1}{2}(\tilde{x}-r_i)^2 + I_{\mathcal{X}}(\tilde{x}) + \rho|\mathcal{N}_i|\tilde{x}^2 - \left(\rho|\mathcal{N}_i|\mathcal{T}_{\mathcal{X}}\left(x_i^k\right) + \rho\sum_{j\in\mathcal{N}_i}\mathcal{T}_{\mathcal{X}}\left(x_j^k\right)\right. \\ &\qquad \left. - \alpha_i^k\right)\tilde{x}, \end{aligned} \tag{4.4}$$

where $I_{\mathcal{X}}(\tilde{x})$ is the indicator function given by

$$I_{\mathcal{X}}(\tilde{x}) = \begin{cases} 0, & \text{if } \tilde{x}\in\mathcal{X}, \\ \infty, & \text{otherwise.} \end{cases}$$

Comparing with the CADMM update in (2.7), we see that (4.4) (and hence (4.2)) together with (4.3) is the CADMM update with $\mathcal{T}_{\mathcal{X}}\left(x_i^k\right)$ and α_i^k as local variables and $\frac{1}{2}(\tilde{x}-r_i)^2 + I_{\mathcal{X}}(\tilde{x})$ as local

objective functions. Then Lemma 2.3 implies that (4.2) and (4.3) lead to

$$\lim_{k\to\infty} \mathcal{T}_{\mathcal{X}}(x_i^k) = \arg\min_{\tilde{x}} \sum_{i=1}^{n} \left(\frac{1}{2}(\tilde{x} - r_i)^2 + I_{\mathcal{X}}(\tilde{x}) \right) = \mathcal{T}_{\mathcal{X}}(\bar{r}).$$

So far we have obtained that BQ-CADMM results from CADMM on a well-solved constrained least-squares problem with the rounding quantization $\mathcal{Q}_d(\cdot)$ applied onto x_i^k; yet we are far from reaching the convergence characterization on BQ-CADMM. When dithered quantizer is used, the expected convergence comes from the linear update of CADMM for average consensus (cf. Eq. (2.9)) as well as the fact that expectation of the output of dithered quantizer is equal to the input. This approach fails to apply to BQ-CADMM because $\mathcal{T}_{\mathcal{X}}(\cdot)$ is not a linear operator in general and one can not simply change the order of projection and expectation. Besides, (4.2) and (4.3) no longer possess the linear convergence rate due to the introduced bounded constraint. As a result, the idea of using the linear convergence rate for DQ-CADMM does not work for BQ-CADMM. On the other hand, if we view BQ-CADMM as a result from the CADMM update on the unconstrained least-squares problem modified by the bounded quantizer $\mathcal{Q}_b(\cdot)$, it is still unclear how BQ-CADMM performs since the quantization error of $\mathcal{Q}_b(\cdot)$ can be unbounded for an unbounded input. Fortunately, α_i^k is inherently bounded by BQ-CADMM update as given below.

Lemma 4.1. *Consider BQ-CADMM with $x_i^0 = 0$ and $\alpha_i^0 = 0$ for $i = 1, 2, \cdots, n$. Then α_i^k is finitely bounded given by*

$$\left|\alpha_i^k\right| \leq (1 + 6\rho|\mathcal{N}_i|)L + |r_i|.$$

Proof. See Section 4.6. □

The bound on α_i^k immediately implies a bound on x_i^k. Now let $\boldsymbol{s}_{\mathcal{Q}_b}^k = [\mathcal{Q}_b(\boldsymbol{x}^k); \boldsymbol{\alpha}^k; \boldsymbol{r}]$. We proceed to rewrite BQ-CADMM as the standard CADMM update on $\boldsymbol{s}_Q^k$ plus an error term caused

by the bounded quantization. We first have

$$\begin{aligned}\mathcal{Q}_b(\boldsymbol{x}^{k+1}) &= \boldsymbol{x}^{k+1} + \boldsymbol{e}_b(\boldsymbol{x}^{k+1}),\\ \boldsymbol{\alpha}^{k+1} &= \boldsymbol{\alpha}^k + \rho \boldsymbol{L}_- \boldsymbol{x}^{k+1} + \rho \boldsymbol{L}_- \boldsymbol{e}_b(\boldsymbol{x}^{k+1}),\end{aligned}$$

where $\boldsymbol{e}_b(\boldsymbol{x}^{k+1}) = \mathcal{Q}_b(\boldsymbol{x}^{k+1}) - \boldsymbol{x}^{k+1}$ is the bounded quantization error. Further defining the following vector $\boldsymbol{s}_e^k = [\boldsymbol{e}_b(\boldsymbol{x}^k); \rho \boldsymbol{L}_- \boldsymbol{e}_b(\boldsymbol{x}^{k+1}); \boldsymbol{0}]$, we have BQ-CADMM update equivalent to

$$\boldsymbol{s}_{\mathcal{Q}_b}^{k+1} = \boldsymbol{D}\boldsymbol{s}_{\mathcal{Q}_b}^k + \boldsymbol{s}_e^{k+1}. \tag{4.5}$$

It is important to note that the above update is deterministic. We will use this fact to establish the main theorem as given below.

Theorem 4.1. *For BQ-CADMM in Algorithm 4.2, there exists a finite time iteration $k_0 > 0$ such that for $k \geq k_0$ all the quantized variable values*

- *either converge to the same quantization value:*

$$\mathcal{Q}_b\left(x_1^k\right) = \cdots = \mathcal{Q}_b\left(x_n^k\right) \triangleq x_{\mathcal{Q}_b}^*,$$

  where $x_{\mathcal{Q}_b}^* \in \Lambda_{\mathcal{X}}$ and

$$\left|x_{\mathcal{Q}_b}^* - \mathcal{T}_{\mathcal{X}}(\bar{r})\right| \leq \left(1 + 4\rho\frac{m}{n}\right)\frac{\Delta}{2}, \tag{4.6}$$

- *or cycle around the average $\bar{r}$ with a finite period $T \geq 2$, i.e., $x_i^k = x_i^{k+T}, i = 1, 2, \ldots, n$. Moreover,*

$$\frac{1}{T}\sum_{l=1}^{T}\mathcal{Q}_b\left(x_1^{k+l}\right) = \cdots = \frac{1}{T}\sum_{l=1}^{T}\mathcal{Q}_b\left(x_n^{k+l}\right) \triangleq \bar{x}_{\mathcal{Q}_b}^*, \tag{4.7}$$

and

$$\left|\bar{x}^*_{\mathcal{Q}_b} - \bar{r}\right| \leq \left(1 + 4\rho\frac{m}{n}\right)\Gamma_0, \text{ where } \Gamma_0 \triangleq \max\left\{\frac{\Delta}{2}, \frac{4\rho n L}{1+2\rho n}\right\}. \tag{4.8}$$

Proof. See Section 4.7. □

Remark 4.1. *Similar to DQ-CADMM's proof, the proof only uses the deterministic scheme and the bounded quantization error of rounding quantizer. Thus, similar consensus results also hold for other deterministic schemes as long as a bounded quantization error is guaranteed, e.g., the rounding down quantizer.*

Remark 4.2. *BQ-CADMM uses $\mathcal{Q}_b\left(x_i^k\right)$ for the $(k+1)$-th update at agent i even though agents can compute and store real values with infinite precision. The reason is to guarantee that $\boldsymbol{\alpha}^k$ is bounded and also lies in $\mathcal{C}(\boldsymbol{L}_-)$. Theorem 4.1 may fail if $\boldsymbol{\alpha}^k \in \mathcal{C}(\boldsymbol{L}_-)$ is not preserved.*

Remark 4.3. *While BQ-CADMM in Algorithm 4.2 starts with variable values $\boldsymbol{x}^0 = \boldsymbol{\alpha}^0 = \mathbf{0}$, one can use similar arguments to show the same result of Theorem 4.1 for any $\boldsymbol{x}^0 \in \mathbb{R}^n$ and $\boldsymbol{\alpha}^0 \in \mathcal{C}(\boldsymbol{L}_-)$. As a result, we can pick $\boldsymbol{x}^0$ and $\boldsymbol{\alpha}^0$ that are respectively closer to the optima $\mathbf{1}_n\bar{r}$ and $\boldsymbol{r} - \mathbf{1}_n\bar{r}$, which usually leads to better consensus values at a fast speed. Note also that Theorem 4.1 holds for any $\rho > 0$. As such, we can use decreasing ρ to accelerate the algorithm while guaranteeing certain consensus accuracy; more details are provided in Section 4.3 where the effect of ρ on BQ-CADMM is discussed.*

Remark 4.4. *With convergence or small enough ρ such that $\Gamma_0 \leq \frac{\Delta}{2}$, the error bound (4.6) and (4.8) are exactly the same as those of DQ-CADMM when $\bar{r} \in \mathcal{X}$. In contrast with [21, 23] where the data need be bounded and the bound is known, BQ-CADMM only requires the data average to lie in $\mathcal{X}$while the agents data can be arbitrary, at a loss of asymptotic convergence to the desired average. This not only saves the energy of data communication but can also induce potential applications; see, e.g., consensus based detection via one-bit communication in Chapter 5.*

While the proposed algorithm is not guaranteed to converge for all cases, we can use the history of agents' variable values (e.g., the running average technique) to reach asymptotic convergence at a consensus. As such, we refer to $x^*_{\mathcal{Q}_b}$ in the convergent case and $\bar{x}^*_{\mathcal{Q}_b}$ in the cyclic case as the resulting consensus value. As a a direct result from Theorem 4.1, the following corollary states that BQ-CADMM must converge when the data average is much beyond the bounded set $\mathcal{X}$.

Corollary 1. *BQ-CADMM must converge to a consensus at a quantization level in $\Lambda_{\mathcal{X}}$ when*

$$||\bar{r}| - L| > \left(1 + 4\rho\frac{n}{m}\right)\Gamma_0.$$

If we further pick $\rho < \frac{n}{4m}$, then $\mathcal{Q}_b(x_i^k)$ must converge to sgn$(\bar{r})L$ where sgn$(\bar{r}) = 1$ if $\bar{r} > 0$ and sgn$(\bar{r}) = -1$ if $\bar{r} < 0$.

The error bounds in Theorem 4.1 also indicate that BQ-CADMM reaches a consensus within a neighborhood of $\bar{r}$. In the case where $\bar{r}$ is far from the set $\mathcal{X}$, however, one will suffer from a large consensus error. Pick ρ small enough such that $\Gamma_0 \leq \frac{\Delta}{2}$ and assume that $L \gg \left(1 + 4\rho\frac{m}{n}\right)\frac{\Delta}{2}$ (e.g, $L \geq 5\left(1 + 4\rho\frac{m}{n}\right)\frac{\Delta}{2}$). Then Corollary 1 implies that $\bar{r}$ is possibly beyond $\mathcal{X}$ when BQ-CADMM converges at $|x^*_{\mathcal{Q}_b}| = L$. Since $\mathcal{T}_{\mathcal{X}}(\bar{r}) = \text{sgn}(\bar{r})L$ and $\left|x^*_{\mathcal{Q}_b} - \mathcal{T}_{\mathcal{X}}(\bar{r})\right| \leq \left(1 + 4\rho\frac{m}{n}\right)\frac{\Delta}{2}$ from (4.6) when $\bar{r} \notin \mathcal{X}$, each agent can subtract sgn$(x^*_{\mathcal{Q}_b})L$ from r_i and hence $\left|\bar{r} - \text{sgn}(x^*_{\mathcal{Q}_b})L\right| = |\bar{r}| - L$. We can then run BQ-CADMM with data $r_i - \text{sgn}(x^*_{\mathcal{Q}_b})L$ and check the resulting convergent value again. We repeat this process until BQ-CADMM converges to $|x^*_{\mathcal{Q}_b}| \neq L$ or reaches a cyclic result. The consensus value at each agent is simply the sum of the final consensus value and the subtracted values from each running. The above is referred to as the extended BQ-CADMM (EBQ-CADMM) presented in Algorithm 4.3.

It is straightforward to see that EBQ-CADMM calls BQ-CADMM at most $\lceil|\bar{r}|/L\rceil + 1$ times. We have the following theorem directly from Theorem 4.1 and Corollary 1.

Theorem 4.2. *Assume that $L \gg \left(1 + 4\rho\frac{m}{n}\right)\frac{\Delta}{2}$. For any agents' data r_i's, EBQ-CADMM in Algorithm 4.3 yields that*

$$t_1 = \cdots = t_n \triangleq t^*,$$

Algorithm 4.3 EBQ-CADMM for quantized consensus

Require: Initialize $t_i = 0$ for each agent $i, i = 1, 2, \cdots, n$. Pick $\rho > 0$ such that $\Gamma_0 \leq \frac{\Delta}{2}$.

1: **while** true **do**
2: Run BQ-CADMM with data r_i at agent i such that either a convergent or cyclic result is achieved. Denote the consensus value as x_{BQ}.
3: **if** $|x_{BQ}| = L$ **then**
4: set $t_i = t_i + \text{sgn}(x_{BQ})L$ and $r_i = r_i - \text{sgn}(x_{BQ})L$
5: **else**
6: break
7: **end if**
8: **end while**
9: **return** $t_i = t_i + x_{BQ}$

and

$$|t^* - \bar{r}| \leq \left(1 + 4\rho\frac{m}{n}\right)\frac{\Delta}{2}.$$

4.3 Effect of Algorithm Parameter

In this section, we discuss the effect of the algorithm parameter ρ on BQ-CADMM and EBQ-CADMM. Since L can be large enough such that $\mathcal{Q}_b(x_i) = \mathcal{Q}_d(x_i)$, the discussion also covers DQ-CADMM using rounding quantizer in Chapter 3 where the effect of ρ is not explored.

4.3.1 Consensus Error

As seen from (4.6) and (4.8), it is clear that the error bound increases in ρ. Therefore, one can pick ρ small enough to achieve a certain consensus accuracy. For example, if $\rho \leq \min\left\{\frac{n}{4m}, \frac{\Delta}{8nL}\right\}$, then the resulting consensus value of EBQ-CADMM is within one quantization resolution of the desired average. Also note that the practical consensus error is usually much smaller than the error bound (see simulations).

4.3.2 Cyclic Period

While the current analysis does not characterize the cyclic period, our simulations in Section 4.3.2 show that BQ-CADMM converges in most cases, particularly with small enough ρ. When the algorithm cycles, the period depends on the network structure, agents' data as well as the algorithm parameter, and varies in $[2, 15]$ of all simulated cases. An important observation is that the cyclic period of all our simulations consists of two consecutive quantization levels for each node. Indeed, we can derive tighter consensus error bounds if it is known *a priori* that the cyclic period only has two consecutive quantization levels, e.g., we can substitute L by Δ for Γ_0 in Thoerem 5.5. See also Chapter 5 where a binary quantizer of two quantization levels is used at each node.

4.3.3 Convergence Time

Here we refer to the smallest k_0 with convergent result or $k_0 + T$ with cyclic result as the convergence time. First notice that it is of no meaning to consider the convergence time without any constraint on the consensus error. To see this, recall that BQ-CADMM has initial variable values $x_i^0 = \alpha_i^0 = 0$ at each agent i and the first iteration has $x_i^1 = \frac{1}{1+2\rho|\mathcal{N}_i|} r_i$ at the first iteration. If ρ is chosen large enough, e.g., $\rho > \frac{1}{2}\max_i r_i$, such that $|x_i^1| < \frac{\Delta}{2}$, then we have $\mathcal{Q}_b(x_i^k) = \alpha_i^k = 0, k = 0, 1, \ldots,$ for all i and the convergence time is $k_0 = 1$. Thus, we consider the selection of ρ to accelerate BQ-CADMM under some consensus accuracy constraint.

Since BQ-CADMM can be viewed as applying the bounded quantization to the unstrained least-squares problem and the quantization error is shown to be bounded, a natural approach is to select ρ that accelerates CADMM. In [50], the optimal parameter is chosen to maximize the convergence rate η given in Theorem 2.2. However, this parameter selection may not meet the consensus accuracy requirement and indeed does provide the best practical performance. Meanwhile, we notice that to compute such ρ requires the knowledge of network structure, which might be unrealistic in large scale networks. Therefore, we propose a heuristic selection $\rho = \frac{n}{m}$ which only requires the number of nodes and the number of edges. Our intuition is based on the fact that

a larger $\frac{m}{n}$ indicates that an agent on the average has more neighboring information to update itself. Therefore, a smaller ρ provides adequate updates towards the consensus. On the contrary, when agents have less information available, a larger ρ can help accelerate the speed. The good performance of this selection is validated by the simulations. Furthermore, in the case where $\rho = \frac{n}{m}$ does not satisfy the consensus accuracy constraint, we come up with a decreasing strategy for parameter selection starting with this ρ such that the convergence time is much shortened.

4.4 Simulations

This section investigates the proposed algorithms via numerical examples. We construct a random connected network in the same way as in Section 3.5. Let $\Delta = 1$ throughout this section.

4.4.1 Consensus Error

We first illustrate how BQ-CADMM and EBQ-CADMM proceed by showing the trajectories of agents' variables. We simulate a connected network with $n = 50$ nodes and $m = 100$ edges. Let $L = \frac{n}{2} = 25$, $\rho = \frac{n}{m} = 0.5$ and $r_i \sim \mathcal{N}(n, n^2)$. Set also the maximum iteration number of each BQ-CADMM to 50. By this setting, it is very likely to have $|\bar{r}| > L$ and we thus apply EBQ-CADMM for data averaging. The trajectories of $\mathcal{Q}_b(x_i^k) + t_i, i = 1, 2, \ldots, n$ are plotted in Fig. 4.1. In this example, the desired average is $\bar{r} = 44.20$ and the resulting consensus value of EBQ-CADMM is $t^* = 44$. The consensus error is $|\bar{r} - t^*| = 0.20$, which is much smaller than the error bound $(1 + 4\rho\frac{m}{n})\frac{\Delta}{2} = 2.5$. The figure also indicates that EBQ-CADMM calls BQ-CADMM twice with the first and second calls converging at the 17-th and 33-th iteration, respectively. As a note, we run the simulation with this setup for $10,000$ times and no cyclic result is observed.

We next compare the proposed algorithms with DQ-CADMM that uses rounding quantizer $\mathcal{Q}_d(\cdot)$ to investigate the effect of the bounded constraint. Denote $\boldsymbol{t} = [t_1; t_2; \cdots; t_n]$ for EBQ-

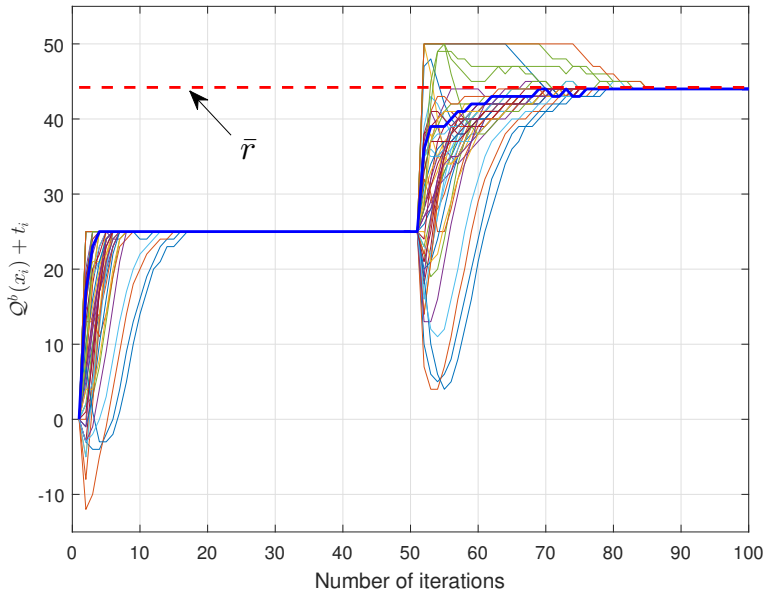


Fig. 4.1: Trajectories of EBQ-CADMM; $n = 50$, $m = 100$, $L = \frac{n}{2}$, $\Delta = 1$, $\rho = 0.5$, and $r_i \sim (n, n^2)$.

CADMM and define the iterative error as

$$\begin{cases} \left\|\mathcal{Q}_b(\boldsymbol{x}^k) - \mathbf{1}_n\bar{r}\right\|_2/\sqrt{n}, & \text{BQ-CADMM}, \\ \left\|\mathcal{Q}_b(\boldsymbol{x}^k) + \boldsymbol{t} - \mathbf{1}_n\bar{r}\right\|_2/\sqrt{n}, & \text{EBQ-CADMM}, \\ \left\|\mathcal{Q}_d(\boldsymbol{x}^k) - \mathbf{1}_n\bar{r}\right\|_2/\sqrt{n}, & \text{DQ-CADMM}, \end{cases}$$

which is equal to the consensus error when a consensus is reached. Set $n = 75$, $m = 200$ and $L = \frac{n}{2}$. Pick $\rho = 0.5$ for CADMM based algorithms and set the maximum iteration number of each BQ-CADMM call in EBQ-CADMM to 50. We consider two cases: one with the average in the bounded set and the other with the average outside the set. Specifically, we generate $r_i \sim \mathcal{N}(0, n^2)$ and run the distributed averaging algorithms twice with data r_i and $r_i + 3n$, respectively. Note that we use BQ-CADMM for the former case since the average lies in the bounded set with high probability. Simulation result is presented in Fig. 4.2. From Fig. 4.2, we see that all the CADMM based algorithms converges to consensuses after certain iterations with consensus errors much smaller than the upper bound 1.83. When $\bar{r} \in \mathcal{X}$, BQ-CADMM performs almost the same as DQ-

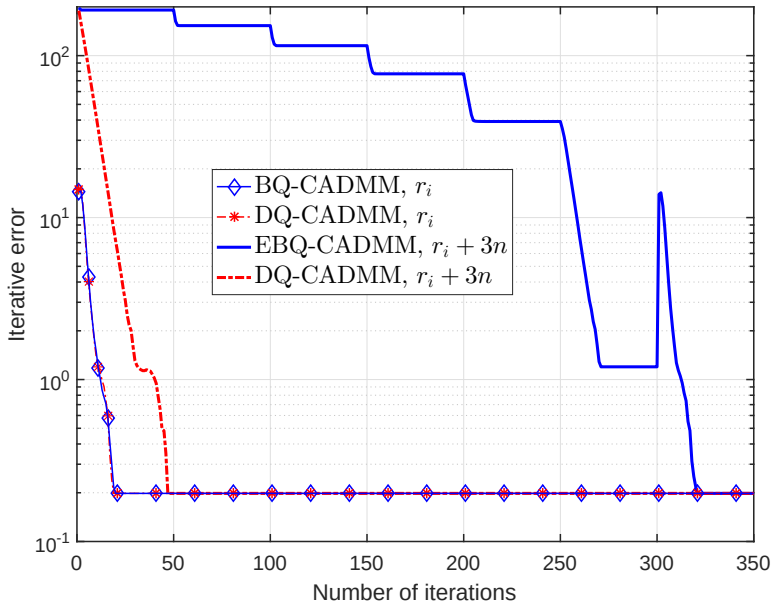


Fig. 4.2: Iterative errors of BQ-CADMM, EBQ-CADMM, and DQ-CADMM. $n = 75$, $m = 200$, $L = \frac{n}{2}$, $\Delta = 1$, $\rho = 0.5$, and $r_i \sim \mathcal{N}(0, n^2)$.

CADMM. When $|\bar{r}| > L$, however, DQ-CADMM converges much faster than EBQ-CADMM. This is because each BQ-CADMM call has iterations running even when a consensus is reached. Also indicated by EBQ-CADMM is that the closer $\bar{r}$ is to the set $\mathcal{X}$, a longer convergence time BQ-CADMM takes.

4.4.2 Cyclic Period

Since the convergent case can be viewed as the cyclic case with period $T = 1$, we first investigate whether BQ-CADMM converges or cycles in various settings. Notice that the quantized consensus algorithm converges in the above examples and indeed, BQ-CADMM and hence EBQ-CADMM tend to converge, particularly with small enough ρ. To illustrate this, we simulate star graphs which have the smallest number of edges for a connected network, randomly generated connected graphs with intermediate numbers of edges, and complete graphs which have the highest number of edges. As Corollary 1 indicates that BQ-CADMM must converge when $\bar{r}$ is far from the bounded set $\mathcal{X}$, we set $L = 30$ and generate $r_i \sim \mathcal{N}(0, 100) + r_0$ where $r_0 \sim \mathcal{N}(0, 25)$ such that most simulated

cases have $\bar{r} \in \mathcal{X}$. We then run BQ-CADMM with different ρ for the same data. The simulation result is plotted in Fig. 4.3 where each plotted value denotes the empirical probability of cyclic result in $10,000$ runs.

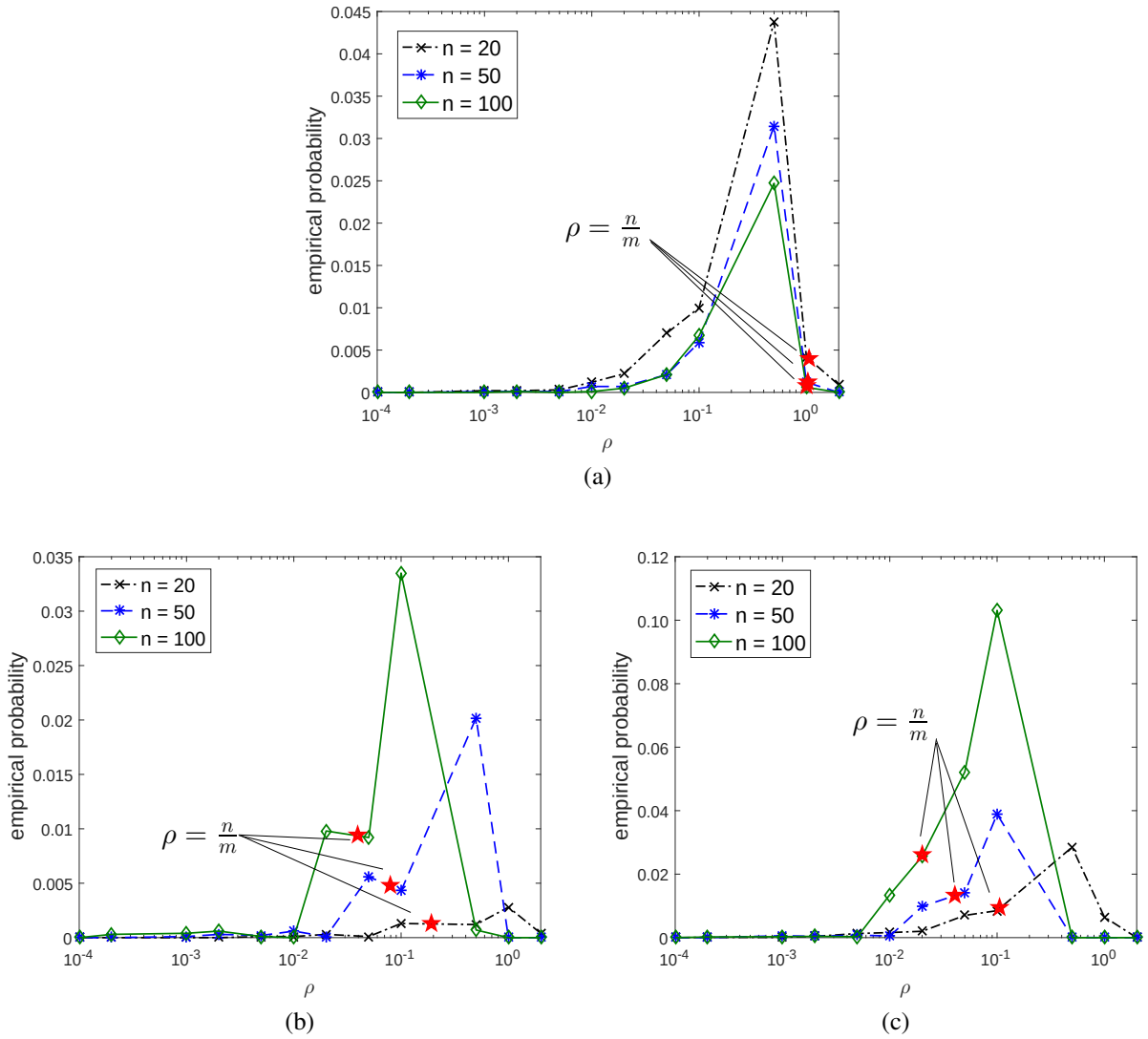


Fig. 4.3: Empirical probability of cyclic case of BQ-CADMM in $10,000$ runs; $r_i \sim \mathcal{N}(0, 100)+r_0$ with $r_0 \sim \mathcal{N}(0, 25)$, $L = 30$ and $\Delta = 1$. (a) star graph, (b) randomly generated graph with $m = \left\lceil \frac{(n+2)(n-1)}{4} \right\rceil$, (c) complete graph.

Fig. 4.3 indicates that BQ-CADMM always converges for large enough ρ with which the consensus result has a large error bound, as discussed in Section 4.3.3. When ρ decreases, the number of cyclic cases first increases and then decreases to 0. In particular, BQ-CADMM converges in all

simulated examples with small enough ρ. Another interesting observation is that peak occurrence of the cyclic case differs for different network structures. The star network has highest cyclic cases around $\rho = 0.5\frac{n}{m}$ and a larger network indicates less cyclic cases, while the intermediate and complete networks have cyclic cases centered around $\rho = 5\frac{n}{m}$ with larger networks having more cyclic cases.

In the same example, we also record the cyclic period when BQ-CADMM indeed cycles. We observe that the period of star networks is always 2 for all n and ρ, and the intermediate and complete networks have periods between 8 and 15. While we cannot draw firm conclusions on cyclic period from the simulation result, we do find that the period in all cyclic cases consists of two consecutive quantization levels for each node, which can help derive better error bounds as discussed in Section 4.3.2.

To study how ρ affects the convergence time of BQ-CADMM, we plot in Fig. 4 the average convergence time of the same example in the above section. We observe that BQ-CADMM converges immediately for large enough ρ but again a large consensus error may exist. The convergence time is about 10 when running BQ-CADMM with $\rho = \frac{n}{m}$ for all n and network structures. For small ρ, the convergence time decreases exponentially as ρ increases. The network structure also plays an important role here. For the star network, the convergence time is almost the same for all n and a larger n results in a slightly longer convergence time. On the other hand, the intermediate and complete networks have longer convergence time with smaller n. Comparing the convergence time of the same network size, we also find that a denser network tends to converge faster for the same ρ

4.4.3 Decreasing Strategy for Parameter Selection

To achieve high consensus accuracy (e.g., the consensus error is within one quantization resolution), the consensus error bound of the cyclic case in Theorem 4.1 implies that one may need to pick a very small ρ which can make BQ-CADMM slow to reach convergence or cycling. Fortunately, simulations indicate that BQ-CADMM converges in most cases and ρ can be larger to meet

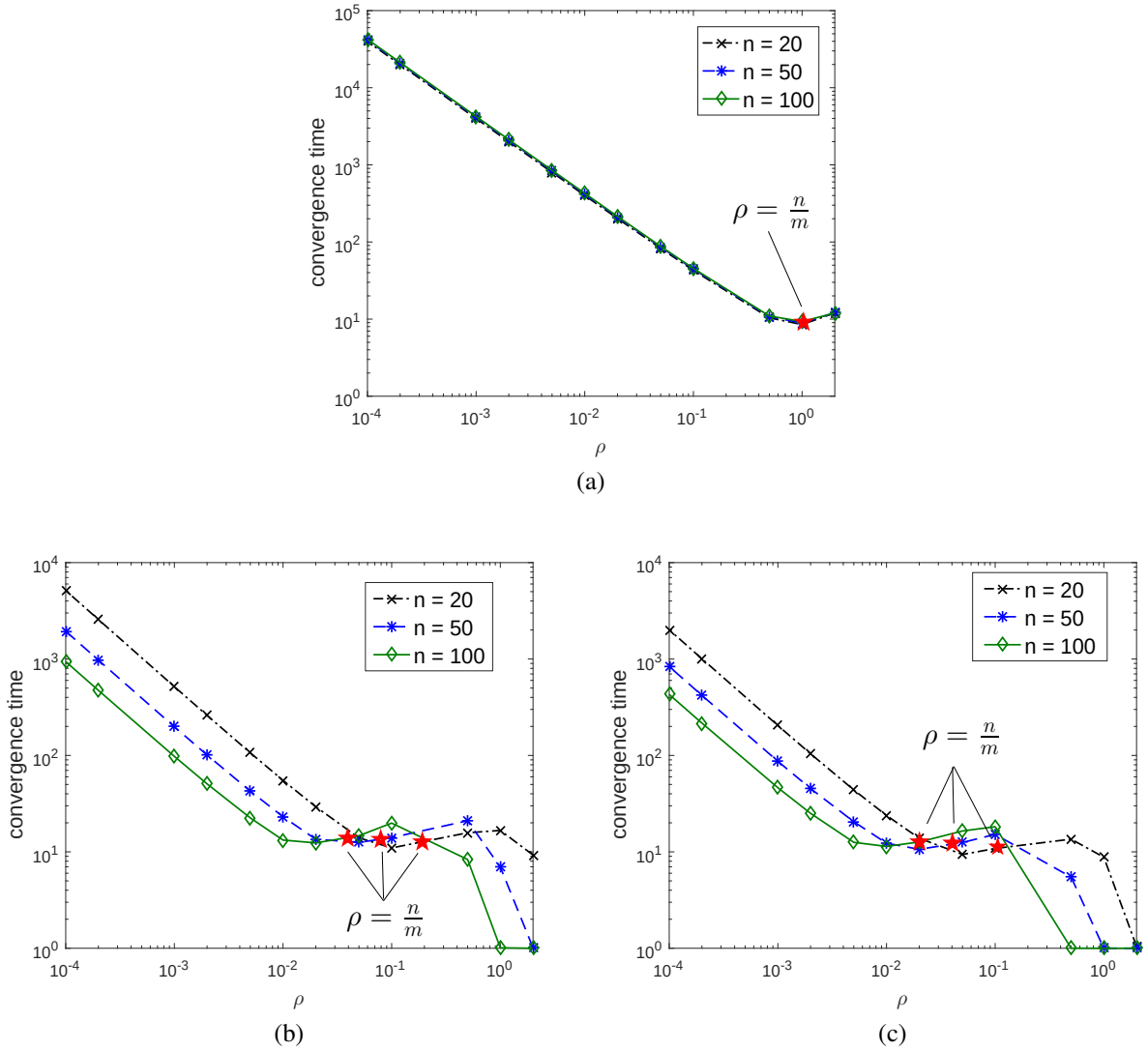


Fig. 4.4: Convergence time of BQ-CADMM where each plotted value is the average of $10,000$ runs; $r_i \sim \mathcal{N}(0, 100) + r_0$ with $r_0 \sim \mathcal{N}(0, 25)$, $L = 30$ and $\Delta = 1$. (a) star graph, (b) randomly generated graph with $m = \left\lceil \frac{(n+2)(n-1)}{4} \right\rceil$, (c) complete graph.

the consensus error requirement in convergent cases. As such, we can start BQ-CADMM with a large ρ for a number of iterations. Even though the local variables may not converge, they become close to the optima. We then pick a small ρ and run BQ-CADMM with the current local variable values. We continue this process until either a convergence is reached and ρ is such that the consensus error bound (4.6) meets the required consensus accuracy, or ρ is small enough such that the cyclic error bound (4.8) satisfies the accuracy requirement. We suggest starting with $\rho = \frac{n}{m}$ as it performs reasonably well in terms of both consensus accuracy and convergence time, as seen from Fig. 4.3 and Fig. 4.4.

To further illustrate the strategy with decreasing step size, we next apply it to the same example in Section 4.3.2. Starting with $\rho = \frac{n}{m}$, we run BQ-CADMM for 50 iterations if $\rho > 10^{-4}$ and then reduce it by a factor of 10. We repeat this process until $\rho < 10^{-4}$ with which we run BQ-CADMM long enough such that either convergence or cycling is reached. The average convergence time of this strategy is presented in Table 4.1. As one can see, this strategy reduces dramatically the convergence time. In addition, most iterations occur when $\rho > 10^{-4}$ and BQ-CADMM only takes a few more iterations to reach the convergence in all simulated cases when $\rho \leq 10^{-4}$.

Table 4.1: Average convergence time of BQ-CADMM with and without using the decreasing strategy for parameter selection.

Structure	Nodes	Decreasing Parameter	Fixed Parameter
Star	20	251.2	3.97×10^4
	50	253.9	4.14×10^4
	100	257.1	4.24×10^4
Intermediate	20	203.9	5.13×10^3
	50	159.2	1.94×10^3
	100	160.6	0.94×10^3
Complete	20	203.3	2.00×10^3
	50	158.7	0.84×10^3
	100	160.4	0.43×10^3

4.5 Summary

This chapter presents quantized consensus algorithms, BQ-CADMM and its extended version EBQ-CADMM, that use finite-bit communications for possibly unbounded data. We provide consensus error bounds to characterize their consensus accuracy and also use numerical examples to examine the convergence time and cyclic behavior. A diminishing step size strategy is then proposed which significantly accelerates the proposed algorithms.

4.6 Proof of Lemma 4.1

Proof. The proof is mainly based on the fact that $|\mathcal{Q}_b(\tilde{x})| \leq L$ for any $\tilde{x} \in \mathbb{R}$. We first prove the following:

$$\mathcal{T}_{\mathcal{X}}(x_i^{k+1}) = \begin{cases} -L, & \text{if } \alpha_i^k > (1+4\rho|\mathcal{N}_i|)\, L + |r_i|\,, \\ L, & \text{if } \alpha_i^k < -(1+4\rho|\mathcal{N}_i|)\, L - |r_i|\,. \end{cases}$$

Assume that $\mathcal{T}_{\mathcal{X}}(x_i^{k+1}) > -L$ if $\alpha_i^k > (1+4\rho|\mathcal{N}_i|)\, L + |r_i|$. Similar to (4.4), the x_i-update of BQ-CADMM implies that $\mathcal{T}_{\mathcal{X}}(x_i^{k+1})$ minimizes a constrained least-squares function:

$$\begin{aligned} \mathcal{T}_{\mathcal{X}}(x_i^{k+1}) &= \arg\min_{\tilde{x}\in\mathcal{X}} \ \frac{1}{2}(\tilde{x}-r_i)^2 + \rho|\mathcal{N}_i|\tilde{x}^2 - \left(\rho|\mathcal{N}_i|\mathcal{Q}_b(x_i^k) + \rho\sum_{j\in\mathcal{N}_i}\mathcal{Q}_b(x_j^k) - \alpha_i^k + r_i\right)\tilde{x} \\ &\triangleq \arg\min_{\tilde{x}\in\mathcal{X}} \ G_i^k(\tilde{x}) + \alpha_i^k\tilde{x}, \end{aligned}$$

where, for ease of presentation, we define

$$G_i^k(\tilde{x}) \triangleq \frac{1}{2}(\tilde{x}-r_i)^2 + \rho|\mathcal{N}_i|\tilde{x}^2 - \left(\rho|\mathcal{N}_i|\mathcal{Q}_b(x_i^k) + \rho\sum_{j\in\mathcal{N}_i}\mathcal{Q}_b(x_j^k) + r_i\right)\tilde{x}.$$

Since $|\mathcal{Q}_b(\tilde{x})| \leq L$ for any $\tilde{x} \in \mathbb{R}$, one can verify that $G_i^k(\tilde{x})$ is Lipschitz continuous over $\mathcal{X}$: for any $\tilde{x}, \tilde{y} \in \mathcal{X}$,

$$\left|G_i^k(\tilde{x}) - G_i^k(\tilde{y})\right| < \left(\left(1 + 4\rho|\mathcal{N}_i|\right) L + |r_i|\right) |\tilde{x} - \tilde{y}| \,.$$

Now letting $\tilde{x} = -L \in \mathcal{X}$, we have

$$\begin{aligned}
&G_i^k(\tilde{x}) + \alpha_i^k(\tilde{x}) - \left(G_i^k\left(\mathcal{T}_{\mathcal{X}}(x_i^{k+1})\right) + \alpha_i^k \mathcal{T}_{\mathcal{X}}(x_i^{k+1})\right) \\
&= G_i^k(-L) - G_i^k\left(\mathcal{T}_{\mathcal{X}}(x_i^{k+1}\right) - \alpha_i^k L - \alpha_i^k \mathcal{T}_{\mathcal{X}}(x_i^{k+1}) \\
&< \left(\left(1 + 4\rho|\mathcal{N}_i|\right) L + |r_i| - \alpha_i^k\right)\left(\mathcal{T}_{\mathcal{X}}(x_i^{k+1}) + L\right), \\
&< 0,
\end{aligned}$$

where the last inequality is because $\alpha_i^k > \left(1 + 4\rho|\mathcal{N}_i|\right) L + |r_i|$ and $\mathcal{T}_{\mathcal{X}}(x_i^{k+1}) > -L$. This contradicts the fact that $\mathcal{T}_{\mathcal{X}}(x_i^{k+1})$ minimizes $G_i^k(\tilde{x}) + \alpha_i^k \tilde{x}$ over $\mathcal{X}$. That $\mathcal{T}_{\mathcal{X}}(x_i^{k+1}) = L$ for $\alpha_i^k < -\left(1 + 4\rho|\mathcal{N}_i|\right) L - |r_i|$ can be shown analogously.

If $\mathcal{T}_{\mathcal{X}}(x_i^{k+1}) = -L$, the bounded quantization scheme implies $\mathcal{Q}_b(x_i^{k+1}) = -L \leq \mathcal{Q}_b(x_j^{k+1})$ for any $j \in \mathcal{N}_i$. Therefore, if $\alpha_i^k > (1 + 4\rho|\mathcal{N}_i|)L + |r_i|$, we get

$$\alpha_i^{k+1} = \alpha_i^k + \rho|\mathcal{N}_i|\mathcal{Q}_b(x_i^{k+1}) - \rho \sum_{j \in \mathcal{N}_i} \mathcal{Q}_b(x_j^{k+1}) \leq \alpha_i^k.$$

Similarly, $\alpha_i^{k+1} \geq \alpha_i^k$ if $\alpha_i^k < -(1 + 4\rho|\mathcal{N}_i|)L - |r_i|$.

Next consider $|\alpha_i^k| \leq (1+4\rho|\mathcal{N}_i|)L+|r_i|$. In this case, we can simply use the triangle inequality to conclude that

$$\begin{aligned}
|\alpha_i^{k+1}| &\leq |\alpha_i^k| + \rho|\mathcal{N}_i| \left|\mathcal{Q}_b(x_i^{k+1})\right| + \rho \sum_{j \in \mathcal{N}_i} \left|\mathcal{Q}_b(x_i^{k+1})\right| \\
&\leq (1 + 6\rho|\mathcal{N}_i|)L + |r_i|.
\end{aligned}$$

Since $\alpha_i^0 = 0$, we finally have $|\alpha_i^k| \leq (1 + 6\rho|\mathcal{N}_i|)L + |r_i|$ for all k. □

4.7 Proof of Theorem 4.1

Proof. We first show that the sequence $\boldsymbol{s}_{\mathcal{Q}_b}^k$ is either convergent or cyclic. Note first that $\mathcal{Q}_b(x_i^k)$ is bounded by L and can only be a multiple of Δ. Similarly, α_i^k is bounded as per Lemma 4.1 and also a multiple of $\rho\Delta$ as seen from the α_i-update of BQ-CADMM. Thus, $\boldsymbol{s}_{\mathcal{Q}_b}^k$ can only have finitely many states as ρ and Δ are both positive. By (4.5) it is clear that BQ-CADMM update of $\boldsymbol{s}_{\mathcal{Q}_b}^{k+1}$ is deterministic and only relies on $\boldsymbol{s}_{\mathcal{Q}_b}^k$. Therefore, $\boldsymbol{s}_{\mathcal{Q}_b}^k$ must be either convergent or cyclic with a finite period $T \geq 2$ after a finite time iteration denoted as k_0. The rest is to use this fact to establish the consensus error bound in the respective case.

Convergent case: In this case we know $\boldsymbol{\alpha}^{k+1} = \boldsymbol{\alpha}^k$ for $k \geq k_0$. Then the $\boldsymbol{\alpha}$-update and Lemma 2.1 implies

$$\mathcal{Q}_b(\boldsymbol{x}^k) \in \mathcal{C}(\mathbf{1}_N),$$

i.e., $\mathcal{Q}_b(\boldsymbol{x}^k)$ reaches a consensus at the same quantization level in $\Lambda_{\mathcal{X}}$. Letting $e_i^* = x_{\mathcal{Q}_b}^* - \mathcal{T}_{\mathcal{X}}(x_i^*)$, we have $|e_i^*| \leq \frac{\Delta}{2}$ since $\mathcal{Q}_b(\cdot) = \mathcal{Q} \circ \mathcal{T}_{\mathcal{X}}(\cdot)$. Now taking $k \to \infty$ on both sides of (4.4) and using the optimality condition for minimizing a convex function (see, e.g., [53]), we get

$$\mathcal{T}_{\mathcal{X}}(x_i^*) - r_i + 2\rho|\mathcal{N}_i|\mathcal{T}_{\mathcal{X}}(x_i^*) + \partial I_{\mathcal{X}}\left(\mathcal{T}_{\mathcal{X}}(x_i^*)\right) - \left(\rho|\mathcal{N}_i|x_{\mathcal{Q}_b}^* + \rho\sum_{j\in\mathcal{N}_i} x_Q^* - \alpha_i^*\right) = 0,$$

where $\partial I_{\mathcal{X}}(\mathcal{T}_{\mathcal{X}}(x_i^*))$ denotes a subgradient of $I_{\mathcal{X}}(\cdot)$ at $\mathcal{T}_{\mathcal{X}}(x_i^*)$. After rearranging and plugging in $\mathcal{T}_{\mathcal{X}}(x_i^*) = x_{\mathcal{Q}_b}^* - e_i^*$, we obtain

$$x_{\mathcal{Q}_b}^* + e_i^* - r_i + \partial I_{\mathcal{X}}\left(x_{\mathcal{Q}_b}^* - e_i^*\right) + \alpha_i^* = 2\sum_{j\in\mathcal{N}_i}|\mathcal{N}_i|e_j^*. \tag{4.9}$$

Summing up both sides of (4.9) from $i = 1$ to n yields

$$\sum_{i=1}^{n} \left(x^*_{\mathcal{Q}_b} + e^*_i - r_i + \partial I_{\mathcal{X}} \left(x^*_{\mathcal{Q}_b} - e^*_i \right) \right) = 2 \sum_{i=1}^{n} |\mathcal{N}_i| e^*_i, \tag{4.10}$$

where $\sum_{i=1}^{N} \alpha^*_i = 0$ is due to $\boldsymbol{\alpha}^* \in \mathcal{C}(\boldsymbol{L}_-)$ and Lemma 2.1. Since $\mathcal{T}_{\mathcal{X}}(\bar{r}) = \arg\min_{\tilde{x}} \sum_{i=1}^{N} \left(\frac{1}{2}(\tilde{x} - r_i)^2 + I_{\mathcal{X}}(\tilde{x}) \right)$, we also have[1]

$$\sum_{i=1}^{N} \left(\mathcal{T}_{\mathcal{X}}(\bar{r}) - r_i + \partial I_{\mathcal{X}} \left(\mathcal{T}_{\mathcal{X}}(\bar{r}) \right) \right) = 0. \tag{4.11}$$

Subtracting (4.11) from (4.10), we get that

$$\sum_{i=1}^{N} \left(\mathcal{T}_{\mathcal{X}}(\bar{r}) - x^*_{\mathcal{Q}_b} + \partial I_{\mathcal{X}} \left(\mathcal{T}_{\mathcal{X}}(\bar{r}) \right) - \partial I_{\mathcal{X}} \left(x^*_{\mathcal{Q}_b} - e^*_i \right) \right) = \sum_{i=1}^{N} \left(2|\mathcal{N}_i| + 1 \right) e^*_i. \tag{4.12}$$

We may only consider the case where $|\mathcal{T}_{\mathcal{X}}(\bar{r}) - x^*_{\mathcal{Q}_b}| > \frac{1}{2}\Delta$ for otherwise (4.6) holds trivially. Recalling that $|e^*_i| \leq \frac{\Delta}{2}$, we get

$$\left(\mathcal{T}_{\mathcal{X}}(\bar{r}) - x^*_{\mathcal{Q}_b} \right) \left(\mathcal{T}_{\mathcal{X}}(\bar{r}) - (x^*_{\mathcal{Q}_b} + e^*_i) \right) \geq 0.$$

Note that the convexity of $I_{\mathcal{X}}(\cdot)$ also implies

$$\left(\partial I_{\mathcal{X}} \left(\mathcal{T}_{\mathcal{X}}(\bar{r}) \right) - \partial I_{\mathcal{X}} \left(x^*_{\mathcal{Q}_b} + e^*_i \right) \right) \left(\mathcal{T}_{\mathcal{X}}(\bar{r}) - (x^*_{\mathcal{Q}_b} + e^*_i) \right) \geq 0.$$

Therefore, $\mathcal{T}_{\mathcal{X}}(\bar{r}) - x^*_{\mathcal{Q}_b}$ and $I_{\mathcal{X}} \left(\mathcal{T}_{\mathcal{X}}(\bar{r}) \right) - \partial I_{\mathcal{X}} \left(x^*_{\mathcal{Q}_b} + e^*_i \right)$ have the same sign and hence the following is true:

$$\left| \mathcal{T}_{\mathcal{X}}(\bar{r}) - x^*_{\mathcal{Q}_b} + \partial I_{\mathcal{X}} \left(\mathcal{T}_{\mathcal{X}}(\bar{r}) \right) - \partial I_{\mathcal{X}} \left(x^*_{\mathcal{Q}_b} + e^*_i \right) \right| \geq \left| \mathcal{T}_{\mathcal{X}}(\bar{r}) - x^*_{\mathcal{Q}_b} \right|.$$

[1]Here $\partial I_{\mathcal{X}} \left(\mathcal{T}_{\mathcal{X}}(\bar{r}) \right)$ denotes a subgradient, i.e., an element from the subdifferential of $I_{\mathcal{X}}(\cdot)$ at $\mathcal{T}_{\mathcal{X}}(\bar{r})$; we simply use the same notation for all the subgradients despite the fact that they might be different values.

Together with (4.12), we can establish the upper bound

$$\left|x^*_{\mathcal{Q}_b} - \mathcal{T}_{\mathcal{X}}(\bar{r})\right| \leq \frac{1}{n}\left|\sum_{i=1}^{n}\left(2\rho|\mathcal{N}_i|+1\right)e_i^*\right| \leq \left(1+\rho\frac{4m}{n}\right)\frac{\Delta}{2},$$

where we use the fact that $\sum_{i=1}^{n}|\mathcal{N}_i| = 2m$ for an undirected connected graph.

Cyclic case: Similar to the convergent case, that $\boldsymbol{s}^k_{\mathcal{Q}_b}$ cycles with a period T implies

$$\boldsymbol{\alpha}^{k+T} - \boldsymbol{\alpha}^k = \rho\boldsymbol{L}_- \sum_{l=1}^{T}\mathcal{Q}_b(\boldsymbol{x}^{k+l}) = \boldsymbol{0},$$

which then leads to (4.7) by Lemma 2.1. While the bound on α_i^k in Lemma 4.1 imposes a bound on x_i^k through the x_i-update, we can derive a tighter bound by utilizing the cyclic fact.

Consider the local variables x_i^k over one period. When $x_i^k \in \mathcal{X}$ for the whole period (and hence for all $k \geq k_0$), we simply have $|e^b(x_i^k)| \leq \frac{\Delta}{2}$. If there is some $x_i^{k+1} > L$ with $k \geq k_0 + T - 1$, then there must exist $k' \leq k$ which is the largest index such that $\mathcal{Q}_b(x_i^{k'}) < L$ and $\mathcal{Q}_b(x_i^{k'+1}) = L$, for otherwise $\mathcal{Q}_b(x_i^k)$ converges and hence BQ-CADMM must converge due to (4.7). Then we have $\mathcal{Q}_b(x_i^{k'+l}) = L$ for $l = 1, 2, \ldots k+1-k'$. Recalling the α_i-update, we get $\alpha_i^{k'+l} \leq \alpha_i^{k'}$ as L is the largest quantization value. We can further write the x_i^{k+1}-update as

$$\begin{aligned}
x_i^{k+1} &= \frac{1}{1+2\rho|\mathcal{N}_i|}\left(\rho|\mathcal{N}_i|\mathcal{Q}_b(x_i^k) + \rho\sum_{j\in\mathcal{N}_i}\mathcal{Q}_b(x_j^k) - \alpha_i^k + r_i\right)\\
&\overset{(a)}{\leq} \frac{1}{1+2\rho|\mathcal{N}_i|}\left(2\rho|\mathcal{N}_i|L - \alpha_i^{k'} + r_i\right)\\
&\overset{(b)}{=} x_i^{k'} + \frac{\rho}{1+2\rho|\mathcal{N}_i|}\Bigg(2|\mathcal{N}_i|L + \sum_{j\in\mathcal{N}_i}\mathcal{Q}_b(x_j^{k'}) - |\mathcal{N}_i|\mathcal{Q}_b(x_i^{k'})\\
&\qquad - |\mathcal{N}_i|\mathcal{Q}_b(x_i^{k'-1}) - \sum_{j\in\mathcal{N}_i}\mathcal{Q}_b(x_j^{k'-1})\Bigg)\\
&\overset{(c)}{\leq} \frac{5\rho|\mathcal{N}_i|L}{1+2\rho|\mathcal{N}_i|} + x_i^{k'} - \frac{\rho|\mathcal{N}_i|}{1+2\rho|\mathcal{N}_i|}\mathcal{Q}_b(x_i^{k'}),
\end{aligned} \tag{4.13}$$

where (a) and (c) are due to fact that $\mathcal{Q}_b(\tilde{x}) \leq L$ for any $\tilde{x} \in \mathbb{R}$, and (b) is from the BQ-CADMM

update at the k'-th iteration. Since k' is the index such that $\mathcal{Q}_b(x_i^{k'}) < L$, we see that (4.13) takes the largest value at $x_i^{k'} = L - \frac{\Delta}{2}$. This further implies for $k \geq k_0$,

$$x_i^k \leq L - \frac{\Delta}{2} + \frac{4\rho|\mathcal{N}_i|L}{1+2\rho|\mathcal{N}_i|} + \frac{\rho|\mathcal{N}_i|\Delta}{1+2\rho|\mathcal{N}_i|} < L + \frac{4\rho|\mathcal{N}_i|L}{1+2\rho|\mathcal{N}_i|}.$$

Similarly, one can show that

$$x_i^k > -L - \frac{4\rho|\mathcal{N}_i|L}{1+2\rho|\mathcal{N}_i|}, \text{ for } k \geq k_0.$$

Therefore, the quantization error satisfies

$$\left|e^b(x_i^k)\right| \leq \max\left\{\frac{\Delta}{2}, \frac{4\rho|\mathcal{N}_i|L}{1+2\rho|\mathcal{N}_i|}\right\} \triangleq \Gamma_0, k \geq k_0.$$

We can now use similar argument in Chapter 3 to derive the error bound. Summing the local variable values over one period and using the above inequality we get

$$\left|\frac{1}{T}\sum_{l=1}^{T}\mathcal{Q}_b(x_i^{k+l}) - \frac{1}{T}\sum_{l=0}^{T}x_i^{k+l}\right| = \left|\bar{x}^*_{\mathcal{Q}_b} - \frac{1}{T}\sum_{l=0}^{T}x_i^{k+l}\right| \leq \Gamma_0. \tag{4.14}$$

By the x_i-update, we can also get

$$\begin{aligned}(1+2\rho|\mathcal{N}_i|)\frac{1}{T}\sum_{l=1}^{T}x_i^{k+l} - \rho|\mathcal{N}_i|\frac{1}{T}\sum_{l=1}^{T}\mathcal{Q}_b(x_i^{k+l}) \\ - \rho\sum_{j\in\mathcal{N}_i}\left(\frac{1}{T}\sum_{l=1}^{T}\mathcal{Q}_b(x_i^{k+l})\right) + \sum_{l=1}^{T}\alpha_i^{k+l} - r_i = 0.\end{aligned}$$

Summing both sides of the above equality from $i = 1$ to n and using (4.7) leads to

$$\sum_{i=1}^{n}\left((1+2\rho|\mathcal{N}_i|)\frac{1}{T}\sum_{l=1}^{T}x_i^{k+l}\right) - \sum_{i=1}^{n}\left(2\rho|\mathcal{N}_i|\frac{1}{T}\sum_{l=1}^{T}\mathcal{Q}_b(x_i^{k+l})\right) + \sum_{i=1}^{n}r_i = 0.$$

Finally, plugging in (4.14) and dividing both sides of the above equation by n, we can bound the

consensus error as

$$\left|\bar{x}^*_{\mathcal{Q}_b} - \bar{r}\right| \leq \left(1 + 4\rho\frac{m}{n}\right)\Gamma_0.$$

This completes the proof. □

CHAPTER 5
CONSENSUS BASED DETECTION IN SENSOR NETWORKS VIA ONE-BIT COMMUNICATIONS

This chapter considers distributed detection over general connected sensor networks using iterative distributed averaging algorithms, with the goal to reach a consensus decision among all sensors in the network. We will show that one-bit communication between linked sensors at each iteration achieves optimal asymptotic performance as centralized settings, which we owe to the use of deterministic quantization scheme.

5.1 Problem and Preliminary

Section 5.1.1 states the problem of this chapter and Section 5.1.2 introduces preliminary concepts and results that will be used to prove our main results in Section 5.3.

5.1.1 Problem Statement

Consider a connected n-node sensor network with m bi-directional links. Each sensor $i, i = 1, 2, \ldots, n$, has its own observation y_i. We model this network as a connected undirected graph with n nodes and m edges. Let $\mathcal{N}_i$ denote the set of directly linked nodes of node i and $|\mathcal{N}_i|$ its cardinality. Here node i is not considered as a linked node of itself, i.e., $i \notin \mathcal{N}_i$. Then $n-1 \leq m \leq \frac{n(n-1)}{2}$, $1 \leq |\mathcal{N}_i| \leq n-1$, and $2m = \sum_{i=1}^{n} |\mathcal{N}_i|$. Assume that the observations y^n originate from an i.i.d. source $Q(y)$ with alphabet Σ that can be either a finite set or any Polish space.[1] Denote $\mathcal{P}(\Sigma)$ as the space of probability measures on Σ. We consider two hypotheses

- $H_1 : Q = P_1 \in \mathcal{P}(\Sigma)$,
- $H_2 : Q = P_2 \in \mathcal{P}(\Sigma)$,

with prior probabilities π_1 and $\pi_2 = 1 - \pi_1$, respectively. Let $\mathcal{A}_n \subseteq \Sigma^n$ denote the acceptance region for H_1 and $\mathcal{A}_n^c = \Sigma^n \setminus \mathcal{A}_n$ the critical region. Then the *type-I and type-II error probabilities* are respectively

$$\alpha_n = P_1(\mathcal{A}_n^c) \text{ and } \beta_n = P_2(\mathcal{A}_n).$$

We investigate the asymptotic detection performance via consensus based approaches where sensors can only reliably exchange one-bit information with its neighbors at each iteration. To ensure autonomy in a large sensor network, local sensors, or more precisely, local computations and communications, do not require the information about the global network structure. We consider the following three criteria for large connected sensor networks:

- Neyman-Pearson criterion with constant constraint: for a given $\alpha \in (0, 1)$,

$$\begin{aligned} \text{maximize} \quad & \liminf_{n\to\infty} -\frac{1}{n} \log \beta_n, \\ \text{subject to} \quad & \lim_{n\to\infty} \alpha_n \leq \alpha. \end{aligned}$$

[1] We follow [54] to use Polish space in order not to be distracted by measurability concerns. In many applications like ours, Σ is either a finite set or a subset of $\mathbb{R}^d$ for some $d \in \mathbb{Z}^+$. Definition of Polish space can be found in [54, pp. 358].

- MAP criterion: given $\pi_1, \pi_2 \in (0,1)$,

$$\text{maximize } \liminf_{n\to\infty} -\frac{1}{n}\log\left(\pi_1\alpha_n + \pi_2\beta_n\right).$$

- Neyman-Pearson criterion with exponential constraint: for a given $\gamma \in (0, D(P_2||P_1))$,

$$\begin{aligned} \text{maximize} \quad & \liminf_{n\to\infty} -\frac{1}{n}\log\beta_n, \\ \text{subject to} \quad & \lim_{n\to\infty} \alpha_n \leq 2^{-n\gamma}. \end{aligned}$$

It is *a priori* unknown a) if exponentially vanishing error probabilities in the network size can be achieved given only one-bit local information exchange; and b) what would be the optimal error exponent if indeed exponentially decaying error probabilities can be attained. Before introducing the consensus based scheme for the construction of acceptance regions, we first review centralized results that act as performance bounds for distributed detection.

5.1.2 Preliminaries

Throughout the rest of this chapter, we assume that $P_1 \in \mathcal{P}(\Sigma)$ and $P_2 \in \mathcal{P}(\Sigma)$ are absolutely mutually continuous. We begin with the definition of relative entropy.

Definition 1. *The relative entropy or Kullback-Leibler divergence between P_1 and P_2 is defined as*

$$D(P_1\|P_2) = \int_\Sigma \log\frac{dP_1}{dP_2}dP_1 = \mathbb{E}_{P_1}\left(\log\frac{dP_1}{dP_2}\right),$$

where dP_1/dP_2 stands for the Radon-Nikodym derivative of P_1 with respect to P_2.

For ease of presentation, we write dP_1 and dP_2 as p_1 and p_2, respectively. Using the weak law of large numbers, we can derive the following asymptotic equipartition property for the relative entropy.

Theorem 5.1 ([55, Theorem 11.8.1])**.** *Let y^n be a sequence of random variables drawn i.i.d. according to P_1, and let P_2 be any other measure from $\mathcal{P}(\Sigma)$. Then*

$$\frac{1}{n}\log\frac{p_1(y^n)}{p_2(y^n)} \to D(P_1\|P_2) \text{ in probability.}$$

Definition 2. *For a fixed n and $\epsilon > 0$, a sequence $y^n \in \Sigma^n$ is said to be relative entropy typical if and only if*

$$D(P_1\|P_2) - \epsilon \leq \frac{1}{n}\log\frac{p_1(y^n)}{p_2(y^n)} \leq D(P_1\|P_2) + \epsilon.$$

The set of relative entropy typical sequences is called the relative entropy typical set $A_\epsilon^{(n)}(P_1\|P_2)$.

We then have the following lemma as a direct consequence of Theorem 5.1.

Lemma 5.1 ([55, Theorem 11.8.2])**.** *Given any positive ϵ,*

$$P_1\left(A_\epsilon^{(n)}(P_1\|P_2)\right) > 1 - \epsilon,$$

provided that n is sufficiently large.

With the above definitions, we are ready to present Stein's lemma which provides the best exponent for type-II error probability under the Neyman-Pearson criterion with constant constraint on type-I error probability.

Theorem 5.2 (Stein's Lemma [55, Theorem 11.8.3], [54, Lemma 3.4.7])**.** *Let y^n be i.i.d. $\sim Q$. Consider the hypothesis between two alternatives $H_1 : Q = P_1$ and $H_2 : Q = P_2$, where $0 < D(P_1\|P_2) < \infty$. Let β_n^α be the infimum of β_n among all tests with $\alpha_n \leq \alpha$. Then for any $0 < \alpha < 1$,*

$$\lim_{n\to\infty} -\frac{1}{n}\log\beta_n^\alpha = D(P_1\|P_2),$$

which can be asymptotically achieved by choosing the acceptance region as $A_\epsilon^{(n)}(P_1\|P_2)$ with $\epsilon \to 0$.

Under the Bayesian framework, Chernoff theorem provides the best error exponent for the error probability.

Theorem 5.3 (Chernoff [55, Theorem 11.9.1], [54, Lemma 3.4.7])**.** *For $\pi_1 > 0$ and $\pi_2 > 0$, the best achievable exponent in the Bayesian probability of error is given by*

$$\liminf_{n\to\infty} -\frac{1}{n}\log(\pi_1\alpha_n + \pi_2\beta_n) = C(P_1, P_2),$$

where $C(P_1, P_2)$ is the Chernoff information defined as

$$C(P_1, P_2) \triangleq -\min_{0\le\lambda\le 1} \log\left(\int_\Sigma p_1(y)^\lambda p_2(y)^{1-\lambda} dy\right).$$

We next present the centralized results under the Neyman-Pearson criterion with exponential constraint via large deviations. Define the logarithmic moment generating function of the LLR as

$$\Lambda(\lambda) = \log \mathbb{E}_{P_1}\left(e^{-\lambda \log \frac{p_1(y)}{p_2(y)}}\right), \lambda \in \mathbb{R}.$$

Notice that $\Lambda(0) = \Lambda(1) = 0$ for the hypothesis testing problem as P_1 and P_2 are mutually absolutely continuous. The Fenchel-Legendre transform of $\Lambda(\lambda)$, which characterizes the large deviations associated with the empirical mean of i.i.d. random variables, is defined as

$$\Lambda^*(\tau) \triangleq \sup_{\lambda\in\mathbb{R}}\{\lambda\tau - \Lambda(\lambda)\}.$$

A useful property of $\Lambda^*(\cdot)$ is stated in the following lemma, which is a direct result of [54, Lemma 2.2.5].

Lemma 5.2. *$\Lambda^*(\tau)$ is a non-decreasing convex function for $\tau > -D(P_1\|P_2)$.*

The following theorem then characterizes the large deviations of the probabilities of error under likelihood ratio tests.

Theorem 5.4 ([54, Theorem 3.4.3])**.** *Let the acceptance region for H_1 be*

$$\left\{ y^n : \frac{1}{n} \log \frac{p_1(y^n)}{p_2(y^n)} > -\tau \right\}. \tag{5.1}$$

Given $\tau \in (-D(P_1\|P_2), D(P_2\|P_1))$, the error probabilities satisfy

$$\lim_{n\to\infty} -\frac{1}{n} \log \alpha_n = \Lambda^*(\tau) > 0,$$

and

$$\lim_{n\to\infty} -\frac{1}{n} \log \beta_n = \Lambda^*(\tau) - \tau > 0.$$

The acceptance region in (5.1) is referred to as Neyman-Pearson test in the literature. It is straightforward to see that the type-II error probability β_n becomes larger as τ increases. Thus, Theorem 5.4 together with the optimality of Neyman-Pearson test (see, e.g., [54, 55]) implies that the optimal error exponent under the Neyman-Pearson criterion with exponential constraint is given by

$$\liminf_{n\to\infty} -\frac{1}{n} \log \beta_n = \Lambda^*(\tau^*) - \tau^*,$$

where τ^* is the smallest value in $(-D(P_1\|P_2), D(P_2\|P_1))$ such that $\Lambda^*(\tau^*) = \gamma$ and must exist for $\gamma \in (0, D(P_2\|P_1))$ as per Stein's lemma and Lemma 5.2. The corresponding acceptance region is then given by the Neyman-Pearson test in (5.1) with $\tau = \tau^*$.

Also shown in [54], both Stein's lemma and Chernoff theorem can be deduced from Theorem 5.4. An interesting fact is that the Chernoff information is equal to the Fenchel-Legendre transform of $\Lambda(\cdot)$ evaluated at zero, i.e., $C(P_1, P_2) = \Lambda^*(0)$. Instead of directly studying the Neyman-Pearson criterion with exponential constraint via large deviations, we will first consider the Neyman-Pearson criterion with constant constraint and the Bayesian criterion in Section 5.3 to help illustrate our approach. In order to apply consensus based approaches, notice that for all the three criteria reviewed in Section 5.1.2, the optimal detectors (i.e., acceptance regions) all amount

to a form of threshold test of the global log-likelihood ratio (LLR). With distributed detection, if one can reconstruct such a statistic, then optimal detection performance in the centralized setting can be attained. Since the global LLR is equivalent to the average of all local LLR values, this motivates the average consensus approach for distributed detection where local LLRs are treated as local agent data. The next section introduces such a distributed averaging algorithm that uses only one-bit quantizer at each node.

5.2 Distributed Average Consensus using One-Bit Communications

The BQ-CADMM approach proposed in Chapter 4 uses a finite-bit quantizer that consists of a projection operator followed by the uniform rounding quantizer. Due to this rounding quantizer, the resulting consensus value is subject to a consensus error from the desired average and the derived error bound has a non-vanishing constant error term (cf. Theorem 4.1). This fact implies that the global LLR thus the consensus based approach is in itself insufficient if one is to attain the same asymptotic performance of the centralized case. We therefore adopt a new binary quantizer with controllable threshold such that the consensus error can be arbitrarily small if consensus is reached at a specific quantization point.

We now construct the one-bit quantizer in a similar fashion to that of Chapter 4: composition of (uniform) quantization and projection. Given quantization resolution $\Delta > 0$ and a predefined quantization point $a \in \mathbb{R}$, let $\mathcal{Q}(\cdot)$ be a uniform quantizer defined as

$$\mathcal{Q}(x) = a + t\Delta, \text{ if } a + t\Delta - \delta < x \leq a + (t+1)\Delta - \delta,$$

where $x \in \mathbb{R}$, $t \in \mathbb{Z}$, and $\delta \in (0, \Delta)$. If we pick $a = 0$ and $\delta = \frac{\Delta}{2}$, then $\mathcal{Q}(\cdot)$ becomes the usual rounding quantizer. Let $\mathcal{X} = [a, a + \Delta]$ and denote by $\mathcal{T}_{\mathcal{X}} : \mathbb{R} \to \mathcal{X}$ the projection operator that

maps $x \in \mathbb{R}$ to the nearest point in $\mathcal{X}$, i.e.,

$$\mathcal{T}_{\mathcal{X}}(x) = \begin{cases} a, & \text{if } x < a, \\ x, & \text{if } a \le x \le a + \Delta, \\ a + \Delta, & \text{otherwise.} \end{cases}$$

The one-bit quantizer is defined as

$$\mathcal{Q}_\delta(\cdot) = \mathcal{Q} \circ \mathcal{T}_{\mathcal{X}}(\cdot), \tag{5.2}$$

which we refer to as δ-quantizer. One can easily verify that the δ-quantizer is equivalent to a binary threshold quantizer

$$\mathcal{Q}_\delta(x) = \begin{cases} a, & \text{if } x \le a + \Delta - \delta, \\ a + \Delta, & \text{otherwise.} \end{cases}$$

Presented in Algorithm 5.4 is BQ-CADMM with this δ-quantizer, where $r_i \in \mathbb{R}$ denotes the local data, i.e., local LLR at node i, and ρ is the algorithm parameter that can be any positive value. It is straightforward to see that BQ-CADMM[2] is fully distributed since the updates of local variables x_i^{k+1} and α_i^{k+1} only rely on its local and neighboring information. While similar results of BQ-CADMM using this δ-quantizer can be obtained as a direct generalization of Theorem 4.1, we can derive tighter consensus error bounds based on the fact that there are only two quantization values. This is stated in Thereom 5.5.

Theorem 5.5. *Let $\bar{r} = \frac{1}{n}\sum_{i=1}^n r_i$ denote the data average. For BQ-CADMM using the δ-quantizer $\mathcal{Q}_\delta(\cdot)$, there exists a finite-time iteration k_0 such that for $k \ge k_0$, all the quantized variable values*

[2]Throughout the rest of this chapter, 'BQ-CADMM' stands for the algorithm with the δ-quantizer; we use 'original BQ-CADMM' to represent the algorithm with the bounded rounding quantizer in Chapter 4

Algorithm 5.4 BQ-CADMM with the δ-quantizer

Require: Initialize $x_i^0 = 0$ and $\alpha_i^0 = 0$ for each agent $i, i = 1, 2, \cdots, n$. Set $\rho > 0$ and $k = 0$.

1: **repeat**

2: every agent i **do**

$$x_i^{k+1} = \frac{1}{1 + 2\rho|\mathcal{N}_i|}\left(\rho|\mathcal{N}_i|\mathcal{Q}_\delta(x_i^k) + \rho\sum_{j\in\mathcal{N}_i}\mathcal{Q}_\delta(x_j^k) - \alpha_i^k + r_i\right),$$

$$\alpha_i^{k+1} = \alpha_i^k + \rho\left(|\mathcal{N}_i|\mathcal{Q}_\delta(x_i^{k+1}) - \sum_{j\in\mathcal{N}_i}\mathcal{Q}_\delta(x_j^{k+1})\right).$$

set $k = k + 1$.

3: **until** a predefined stopping criterion (e.g., a maximum iteration number) is satisfied.

either

- *converge to the same quantization value:*

$$\mathcal{Q}_\delta(x_1^k) = \mathcal{Q}_\delta(x_2^k) = \cdots = \mathcal{Q}_\delta(x_n^k) \triangleq x_{\mathcal{Q}_\delta}^* \in \{a, a + \Delta\},$$

where $x_{\mathcal{Q}_\delta}^*$ satisfies the following error bound

$$\left|x_{\mathcal{Q}_\delta}^* - \mathcal{T}_{\mathcal{X}}(\bar{r})\right| \leq \begin{cases} \left(1 + 4\rho\frac{m}{n}\right)(\Delta - \delta), & \text{if } x_{\mathcal{Q}_\delta}^* = a, \\ \left(1 + 4\rho\frac{m}{n}\right)\delta, & \text{if } x_{\mathcal{Q}_\delta}^* = a + \Delta, \end{cases} \tag{5.3}$$

- *or cycle around the true average $\bar{r}$ with a finite period $T \geq 2$, i.e., $x_i^k = x_i^{k+T}, i = 1, 2, \cdots, n$. Furthermore,*

$$\sum_{l=0}^{T-1}\mathcal{Q}_\delta(x_1^{k+l}) = \sum_{l=0}^{T-1}\mathcal{Q}_\delta(x_2^{k+l}) = \cdots = \sum_{l=0}^{T-1}\mathcal{Q}_\delta(x_n^{k+l}), \tag{5.4}$$

and

$$|\bar{r} - (a + \Delta - \delta)| < 6\rho n\Delta. \tag{5.5}$$

Proof. See Section 5.7. □

Remark 5.1. *Contrasting with original BQ-CADMM that has a uniform consensus error bound when the algorithm converges, using the δ-quantizer results in error bounds that are dependent on the consensus value. This is achieved by the asymmetric rounding of the δ-quantizer. Clearly, choosing a small δ relative to $\frac{\Delta}{2}$ will skew the quantizer toward 0. Thus, when consensus is reached at $a + \Delta$, the consensus error is ensured to be small too.*

Remark 5.2. *While Theorem 5.5 only requires a connected network, the convergence time (the smallest k_0 in convergent cases or the smallest $k_0 + T$ in cyclic cases) depends on the agents' data, the network structure as well as the algorithm parameter ρ. Besides, BQ-CADMM converges in most cases, particularly with large and dense networks or small enough algorithm parameters, as shown in Chapter 4.*

Remark 5.3. *It is worth noting the choice of the algorithm parameter ρ. For BQ-CADMM to work, i.e., Theorem 5.5 to hold, ρ can be any positive value and does not depend on other parameters; however, to guarantee certain accuracy as we will do in the next section, ρ has to be selected according to related quantities such as number of nodes and number of edges. In addition, the choice of ρ has an impact on whether convergence or oscillation can happen. To see this, consider $\bar{r} \neq a + \Delta - \delta$. Then (5.5) is violated with small enough ρ and convergence must be reached. This may also explain why small algorithm parameters are likely to yield convergence results in our simulations where $\bar{r}$ itself is random.*

5.3 Optimal Asymptotic Performance

This section establishes the optimal asymptotic performance under the three criteria. We use the consensus result from BQ-CADMM with appropriate algorithm parameter and quantizer setup to construct acceptance regions that can asymptotically achieve the optimal performance in centralized settings.

5.3.1 Neyman-Pearson Criterion with Constant Constraint

In Stein's lemma, the relative entropy typical set $A_\epsilon^{(n)}(P_1\|P_2)$ can achieve the optimal error exponent in the centralized setting with diminishing ϵ. Consequently, by picking suitable ρ and small enough δ we can construct an acceptance region that is similar to $A_\epsilon^{(n)}(P_1\|P_2)$ to achieve the same error exponent. The result is presented in the following theorem..

Theorem 5.6. *Let y^n be i.i.d. $\sim Q$. Consider the hypothesis between two alternatives, $H_1 : Q = P_1$ and $H_2 : Q = P_2$, where $0 < D(P_1\|P_2) < \infty$. Let $r_i = \log \frac{p_1(y_i)}{p_2(y_i)}$ be the local data at node i. Set $a = 0$, $\Delta = D(P_1\|P_2)$, and*

$$\rho = \min\left\{\frac{\delta}{6nD(P_1\|P_2)}, \frac{n}{4m}\right\}.$$

Assume that BQ-CADMM runs sufficiently long such that either convergence or cycling occurs. Let the acceptance region for H_1 be

$$\mathcal{A}_n^\delta = \{y^n : \textit{BQ-CADMM cycles}\} \bigcup \{y^n : \textit{BQ-CADMM converges at } x^*_{\mathcal{Q}_\delta} = D(P_1\|P_2)\}.$$

Then given any $0 < \delta < D(P_1\|P_2)$,

$$P_1\left(\left(\mathcal{A}_n^\delta\right)^c\right) < \frac{1}{2}\delta, \textit{ for } n \textit{ sufficiently large}. \tag{5.6}$$

Moreover,

$$\lim_{\delta\to 0}\lim_{n\to\infty} -\frac{1}{n}\log P_2\left(\mathcal{A}_n^\delta\right) = D(P_1\|P_2). \tag{5.7}$$

Proof. In this case we have $\bar{r} = \frac{1}{n}\log\frac{p_1(y^n)}{p_2(y^n)}$. We first find a *sufficient* condition for $y^n \in \mathcal{A}_n^\delta$ to establish (5.6). If $y^n \notin \mathcal{A}_n^\delta$, then BQ-CADMM converges at $x^*_{\mathcal{Q}_\delta} = 0$ and hence (5.3) implies

$$\left|\mathcal{T}_{\mathcal{X}}\left(\frac{1}{n}\log\frac{p_1(y^n)}{p_2(y^n)}\right)\right| \le \left(1+4\rho\frac{m}{n}\right)(D(P_1\|P_2)-\delta).$$

Picking $0 < \rho \le \rho_1 \triangleq \frac{n\delta}{9m(D(P_1\|P_2)-\delta)}$, we have

$$\left(1+4\rho\frac{m}{n}\right)(D(P_1\|P_2)-\delta) \le D(P_1\|P_2) - \frac{5}{9}\delta.$$

Thus, if y^n is such that $\left|\frac{1}{n}\log\frac{p_1(y^n)}{p_2(y^n)} - D(P_1\|P_2)\right| \le \frac{1}{2}\delta$, y^n must lie in $\mathcal{A}_n^\delta$ as $\mathcal{T}_{\mathcal{X}}(\cdot)$ projects a real value to the nearest point in $\mathcal{X} = [0, D(P_1\|P_2)]$. Therefore,

$$\mathcal{A}_n^\delta \supseteq \left\{y^n : \left|\frac{1}{n}\log\frac{p_1(y^n)}{p_2(y^n)} - D(P_1\|P_2)\right| \le \frac{1}{2}\delta\right\} = A^{(n)}_{\frac{1}{2}\delta}(P_1\|P_2).$$

Hence, (5.6) is true according to Lemma 5.1.

We next show that $\mathcal{A}_n^\delta$ can asymptotically achieve the optimal error exponent by identifying a *necessary* condition for $y^n \in \mathcal{A}_n^\delta$. When convergence happens, $x^*_{\mathcal{Q}_\delta} = D(P_1\|P_2)$ and (5.3) implies that

$$\left|\mathcal{T}_{\mathcal{X}}\left(\frac{1}{n}\log\frac{p_1(y^n)}{p_2(y^n)}\right) - D(P_1\|P_2)\right| \le \left(1+4\rho\frac{m}{n}\right)\delta.$$

If we again pick ρ small enough, e.g., $0 < \rho \le \rho_2 \triangleq \frac{n}{4m}$, then $\left(1+4\rho\frac{m}{n}\right)\delta \le 2\delta$. Thus,

$$\mathcal{T}_{\mathcal{X}}\left(\frac{1}{n}\log\frac{p_1(y^n)}{p_2(y^n)}\right) \ge D(P_1\|P_2) - 2\delta.$$

Recall the definition of $\mathcal{T}_{\mathcal{X}}(\cdot)$, we have

$$\frac{1}{n}\log\frac{p_1(y^n)}{p_2(y^n)} \geq D(P_1\|P_2) - 2\delta. \tag{5.8}$$

Now if BQ-CADMM cycles, we have from Theorem 5.5 that

$$\left|\frac{1}{n}\log\frac{p_1(y^n)}{p_2(y^n)} - (D(P_1\|P_2) - \delta)\right| < 6\rho n\Delta.$$

Letting $\rho \leq \rho_3 = \frac{\delta}{6nD(P_1\|P_2)}$, we conclude that (5.8) is also true. Thus, if $y^n \in \mathcal{A}_n^\delta$, we have

$$p_2(y^n) \leq p_1(y^n)2^{-n(D(P_1\|P_2)-2\delta)}.$$

Therefore,

$$\begin{aligned}
-\frac{1}{n}\log P_2(\mathcal{A}_n^\delta) &= -\frac{1}{n}\log\int_{\mathcal{A}_n^\delta} p_2(y^n)dy \\
&\geq -\frac{1}{n}\log\left(2^{-n(D(P_1\|P_2)-2\delta)}\int_{\mathcal{A}_n^\delta} p_1(y^n)dy\right) \\
&\geq D(P_1\|P_2) - 2\delta,
\end{aligned}$$

which, together with Stein's lemma, implies (5.7).

The proof is complete by choosing $\rho = \min\{\rho_1, \rho_2, \rho_3\}$ and noting that $\delta < D(P_1\|P_2)$ and $m \leq \frac{n(n-1)}{2}$. □

Therefore, by choosing small enough δ, we have that $\alpha_n < \frac{1}{2}\delta \leq \alpha$ for large n and that the type-II error exponent is arbitrarily close to the optimal error exponent $D(P_1\|P_2)$. Moreover, the above proof implies that as long as $\delta \to 0$ with $n \to \infty$, we can get

$$\lim_{n\to\infty} -\frac{1}{n}\log\beta_n = D(P_1\|P_2). \tag{5.9}$$

On the other hand, δ cannot decrease too fast in order to satisfy the type-I error constraint. With

finite alphabet, an option is from Hoeffding's test [56] by setting $\delta = \frac{|\Sigma| \log n}{n}$ where $|\Sigma|$ denotes the cardinality of the alphabet. In general, there is not such a universal selection of δ. Fortunately, we can always calculate δ such that $P_1\left(A^{(n)}_{\frac{1}{2}\delta}\right) = 1-\alpha$, and it is not hard to show that this δ diminishes in n. If the above choice is greater than or equal to $D(P_1\|P_2)$, we can simply set $\delta = \frac{D(P_1\|P_2)}{2}$ to ensure $0 < \delta < D(P_1\|P_2)$. In this way, (5.9) is guaranteed under the type-I error constraint.

5.3.2 MAP Criterion

Unlike the Neyman-Pearson criterion with constant constraint, the MAP criterion does not require δ to be diminishing. As one will see, this is because the optimal acceptance region converges to the same set asymptotically for any positive prior probabilities. Specifically, it is well-known that the optimal acceptance region for any n under the MAP criterion is

$$\left\{y^n : \frac{1}{n}\log\frac{p_1(y^n)}{p_2(y^n)} > \frac{1}{n}\log\frac{\pi_2}{\pi_1}\right\}, \tag{5.10}$$

provided that π_1 and π_2 are both positive. Chernoff theorem thus indicates that

$$\begin{aligned}
&\liminf_{n\to\infty} -\frac{1}{n}\log P_1\left(\left\{y^n : \frac{1}{n}\log\frac{p_1(y^n)}{p_2(y^n)} \le \frac{1}{n}\log\frac{\pi_2}{\pi_1}\right\}\right)\\
=&\liminf_{n\to\infty} -\frac{1}{n}\log P_2\left(\left\{y^n : \frac{1}{n}\log\frac{p_1(y^n)}{p_2(y^n)} > \frac{1}{n}\log\frac{\pi_2}{\pi_1}\right\}\right)\\
=&\, C(P_1, P_2).
\end{aligned} \tag{5.11}$$

We remark that (5.11) does not depend on particular values of π_1 and π_2 as long as they are positive. The following theorem states the best achievable exponent for the Bayesian error probability.

Theorem 5.7. *Let y^n be i.i.d $\sim Q$. Consider the hypothesis between $H_1 : Q = P_1$ and $H_2 : Q = P_2$ with positive prior probabilities π_1 and π_2, respectively. For the δ-quantizer $\mathcal{Q}_\delta(\cdot)$, set $a = -1$, $\Delta = 2$, and $\delta = 1$. Set also the local data $r_i = \log\frac{p_1(y_i)}{p_2(y_i)}$ and the algorithm parameter $\rho = \frac{1}{12n^2}$. Assume that BQ-CADMM runs sufficiently long such that either the convergence or cycling occurs.*

Let the acceptance region for H_1 be

$$\mathcal{A}_n = \{y^n : \textit{BQ-CADMM cycles}\} \bigcup \{y^n : \textit{BQ-CADMM converges at } x^*_{\mathcal{Q}_\delta} = 1\}.$$

Then the error exponent is given by

$$\liminf_{n\to\infty} -\frac{1}{n}\log(\pi_1\alpha_n + \pi_2\beta_n) = C(P_1, P_2).$$

Proof. With this setup, the δ-quantizer has the threshold at $a + \Delta - \delta = 0$. First note that if BQ-CADMM results in oscillation, Theorem 5.5 implies

$$\left|\frac{1}{n}\log\frac{p_1(y^n)}{p_2(y^n)}\right| < 6\rho n\Delta = \frac{1}{n},$$

where the last inequality is because $\rho = \frac{1}{12n^2}$.

In the convergent case, we first use a *necessary* condition for $x^*_{\mathcal{Q}_\delta} = 1$ to show that

$$\liminf_{n\to\infty} -\frac{1}{n}\log\beta_n \geq C(P_1, P_2).$$

By Theorem 5.5, when $x^*_{\mathcal{Q}_\delta} = 1$, y^n must satisfy

$$\left|\mathcal{T}_{\mathcal{X}}\left(\frac{1}{n}\log\frac{p_1(y^n)}{p_2(y^n)}\right) - 1\right| < 1 + 4\rho\frac{m}{n}.$$

If we pick $\rho \leq \frac{1}{4m}$ and recall that $\mathcal{T}_{\mathcal{X}}(\cdot)$ is the projection operator that maps a real value to the nearest point in $\mathcal{X} = [-1, 1]$, the above inequality indicates that y^n is such that

$$\frac{1}{n}\log\frac{p_1(y^n)}{p_2(y^n)} > -\frac{1}{n}.$$

Together with the cyclic case, we get

$$\mathcal{A}_n \subseteq \left\{ y^n : \frac{1}{n} \log \frac{p_1(y^n)}{p_2(y^n)} > -\frac{1}{n} \right\}.$$

Hence,

$$\beta_n = P_2(\mathcal{A}_n) \leq P_2 \left(\left\{ y^n : \frac{1}{n} \log \frac{p_1(y^n)}{p_2(y^n)} > -\frac{1}{n} \right\} \right).$$

Comparing with (5.10), we see that $\{y^n : \frac{1}{n} \log \frac{p_1(y^n)}{p_2(y^n)} > -\frac{1}{n}\}$ is the optimal acceptance region for the hypothesis testing problem with prior probabilities $\frac{2}{3}$ and $\frac{1}{3}$ under H_1 and H_2, respectively. Therefore, (5.11) implies that

$$\begin{aligned} &\liminf_{n\to\infty} -\frac{1}{n} \log \beta_n \\ \geq &\liminf_{n\to\infty} -\frac{1}{n} \log P_2 \left(\left\{ y^n : \frac{1}{n} \log \frac{p_1(y^n)}{p_2(y^n)} > -\frac{1}{n} \right\} \right) \\ = &\, C(P_1, P_2). \end{aligned} \tag{5.12}$$

We next find a *sufficient* condition for $y^n \in \mathcal{A}_n$ to establish

$$\liminf_{n\to\infty} -\frac{1}{n} \log \alpha_n \geq C(P_1, P_2). \tag{5.13}$$

When $y^n \notin \mathcal{A}_n$, convergence must be reached with $x^*_{\mathcal{Q}_\delta} = -1$ and we have

$$\left| \mathcal{T}_{\mathcal{X}} \left(\frac{1}{n} \log \frac{p_1(y^n)}{p_2(y^n)} \right) + 1 \right| \leq 1 + 4\rho \frac{m}{n}.$$

Therefore, when $n \geq 2$ and $\rho \leq \frac{1}{4m}$,

$$\mathcal{A}_n^c \subseteq \left\{ y^n : \frac{1}{n} \log \frac{p_1(y^n)}{p_2(y^n)} \leq \frac{1}{n} \right\}.$$

Since $\left\{ y^n : \frac{1}{n} \log \frac{p_1(y^n)}{p_2(y^n)} > \frac{1}{n} \right\}$ is the optimal acceptance region for the hypothesis testing problem with prior probabilities $\frac{1}{3}$ and $\frac{2}{3}$, (5.13) can be shown similarly.

Finally, combining (5.12) and (5.13) we have

$$\liminf_{n\to\infty} -\frac{1}{n}\log\left(\pi_1\alpha_n + \pi_2\beta_n\right) \geq C(P_1, P_2).$$

The proof is complete by Chernoff theorem and the fact that $m \leq \frac{n(n-1)}{2}$ for a connected undirected graph. □

Remark 5.4. *It appears that choosing $\rho = \frac{1}{12n^2}$, which can be very small, may make BQ-CADMM slow. Fortunately, BQ-CADMM is more likely to converge with larger n and we only need $\rho \leq \frac{1}{4m}$ if convergence happens. The decreasing strategy in Chapter 4 for ρ can also be used to accelerate the speed of BQ-CADMM.*

The above theorem indicates that the consensus approach achieves the optimal error exponent which is given by Chernoff theorem. A direct extension is to consider multi-hypothesis testing. We will show that our consensus based approach can also achieve the centralized error exponent under MAP criterion if BQ-CADMM are run multiple times. Under the w-th hypothesis, denote the probability measure by P_w and prior probability by π_w, where $w = 1, 2, \cdots, W$ for some integer $W \geq 2$. We also denote dP_w as p_w. Assume that all π_w are positive and that P_w and $P_{w'}$ are absolutely mutually continuous for any $w \neq w'$. The centralized MAP rule for the w-th hypothesis is given by

$$A_w^*(n) = \left\{ y^n : \pi_w p_w(y^n) \geq \max_{w'<w} \pi_{w'} p_{w'}(y^n), \pi_w p_w(y^n) > \max_{w'>w} \pi_{w'} p_{w'}(y^n) \right\}.$$

For ease of presentation, define the following Neyman-Pearson test between two different hypotheses

$$V_n^{w,w'} = \begin{cases} \left\{ y^n : \frac{1}{n}\sum_i \log\frac{p_w(y^i)}{p_{w'}(y^i)} \geq \frac{1}{n}\log\frac{\pi_{w'}}{\pi_w} \right\}, \text{ if } w' < w, \\ \left\{ y^n : \frac{1}{n}\sum_i \log\frac{p_w(y^i)}{p_{w'}(y^i)} > \frac{1}{n}\log\frac{\pi_{w'}}{\pi_w} \right\}, \text{ if } w' > w. \end{cases}$$

Then we can write $A_w^*(n) = \bigcup_{w'\neq w} V_n^{w,w'}$ for a given w. Two useful facts about $V_n^{w,w'}$ are stated as follows:

- $V_n^{w,w'} = \Sigma^n \setminus V_n^{w',w} \triangleq \left(V_n^{w',w}\right)^c$.
- Consider binary hypothesis testing between P_w and P'_w with prior probabilities $\frac{\pi_w}{\pi_w+\pi_{w'}}$ and $\frac{\pi_{w'}}{\pi_w+\pi_{w'}}$, respectively. Then $V_n^{w,w'}$ is the optimal acceptance region for P_w under MAP criterion. Thus, Chernoff theorem implies

$$\liminf_{n\to\infty} -\frac{1}{n}\log P_w\left(\left(V_n^{w,w'}\right)^c\right) = \liminf_{n\to\infty} -\frac{1}{n}\log P_{w'}\left(V_n^{w,w'}\right) = C(P_w, P_{w'}).$$

Now consider the optimal Bayesian error for the multi-hypothesis testing

$$P_e^* = \sum_w \pi_w P_w((A_w^*(n))^c).$$

Noting that $P_w((A_w^*(n))^c) = P_w\left(\bigcup_{w'\neq w}\left(V_n^{w,w'}\right)^c\right)$, we get the following

$$\max_{w'\neq w} P_w\left(\left(V_n^{w,w'}\right)^c\right) \leq P_w((A_w^*(n))^c) \leq \sum_{w'\neq w} P_w\left(\left(V_n^{w,w'}\right)^c\right). \tag{5.14}$$

Thus, we get the error exponent

$$\liminf_{n\to\infty} -\frac{1}{n}\log P_w((A_w^*(n))^c) = \min_{w'\neq w} C(P_w, P_{w'}),$$

where w is prefixed. Hence, the centralized error exponent is

$$\liminf_{n\to\infty} -\frac{1}{n}\log P_e^* = \min_w \min_{w'\neq w} C(P_w, P_{w'}).$$

To apply our consensus based approach, we next use the bubble sorting idea to construct the MAP detector: starting with $w = 1$ and $w' = 2$, test whether $y^n \in V_n^{w,w'}$; if yes, keep this w, and otherwise, set $w = w'$; increment $w' = w' + 1$ and test again if $y^n \in V_n^{w,w'}$; continue this process until the W-th hypothesis is involved. It is straightforward to see that the final w is the output of the MAP detector. Recall that the acceptance region in Theorem 5.7, when testing between

P_w and $P_{w'}$, achieves the same optimal error exponent as $V_n^{w,w'}$. We may replace $V_n^{w,w'}$ with this acceptance region, denoted by $\mathcal{A}_n^{w,w'}$, to implement the consensus based approach. In summary, the above algorithm runs BQ-CADMM $W-1$ times to make a decision for the multi-hypothesis testing problem.

To study how this algorithm performs, let $\mathcal{A}_w(n)$ be the acceptance region for the w-th hypothesis resulting from the $W-1$ runs of BQ-CADMM. For the first hypothesis to be selected, y^n must be such that $y^n \in \bigcap_{w'\neq 1} \mathcal{A}_n^{1,w'}$ and conversely, if $y^n \in \bigcap_{w'\neq 1} \mathcal{A}_n^{1,w'}$, we must select the first hypothesis. Thus, $\mathcal{A}_1(n) = \bigcap_{w'\neq 1} \mathcal{A}_n^{1,w'}$. Similar to (5.14) and using Theorem 5.7, we have

$$\liminf_{n\to\infty} -\frac{1}{n}\log P_1((\mathcal{A}_1(n))^c) = \min_{w'\neq 1} C(P_1, P'_w).$$

For $2 \leq w \leq W-1$, if the final decision is w, then the $(w-1)$-th run of BQ-CADMM must accept w. At the same time, y^n must lie in $\mathcal{A}_n^{w,w'}$ for any $w' > w$. Thus,

$$\left(\bigcap_{w'<w} (A_n^{w',w}(n))^c\right) \bigcap \left(\bigcap_{w'>w} \mathcal{A}_n^{w,w'}\right) \subset \mathcal{A}_w(n) \subset \bigcap_{w'>w} \mathcal{A}_n^{w,w'}.$$

Recalling the definition of $\mathcal{A}_n^{w,w'}$, we conclude that

$$\min_{w'\neq w} C(P_w, P_{w'}) \leq \liminf_{n\to\infty} -\frac{1}{n}\log P_w((\mathcal{A}_w(n))^c) \leq \min_{w'>w} C(P_w, P_{w'}).$$

For the W-th hypothesis, similar argument shows that

$$\min_{w'\neq W} C(P_W, P_{w'}) \leq \liminf_{n\to\infty} -\frac{1}{n}\log P_W((\mathcal{A}_W(n))^c) \leq \max_{w'\neq W} C(P_W, P_{w'}).$$

Finally, it is noted that the Bayesian error is $P_e = \sum_w \pi_w P_w((\mathcal{A}_w(n))^c)$ and that its error exponent $\liminf_{n\to\infty} -\frac{1}{n}\log P_e$ is decided by the lowest error exponent of $P_w((\mathcal{A}_w(n))^c)$. Together with the

symmetry property of $C(P_w, P_{w'})$ (i.e., $C(P_w, P_{w'}) = C(P_{w'}, P_w)$), we conclude that

$$\liminf_{n\to\infty} -\frac{1}{n}\log P_e = \min_{w}\min_{w'\neq w} C(P_w, P_{w'}). \tag{5.15}$$

which is the optimal error exponent in centralized case. The result is summarized in the following theorem.

Theorem 5.8. *Consider multi-hypothesis testing with hypotheses $P_w, w = 1, 2, \cdots, W$ for some integer $W \geq 2$. Assume that the prior probability π_w for each hypothesis is positive and that the hypotheses P_w are absolutely mutually continuos with each other. Using $W-1$ runs of BQ-CADMM with the same quantizer and algorithm parameters in Theorem 5.7, each node can achieve the optimal centralized error exponent under the MAP criterion, which is given in (5.15).*

5.3.3 Neyman-Pearson Criterion with Exponential Constraint

We now consider the Neyman-Pearson criterion with exponential constraint based on large deviations techniques. Similar to the above two cases, the key is to pick appropriate algorithm and quantizer parameters such that the constructed acceptance region approaches the optimal centralized one asymptotically.

Theorem 5.9. *Let y^n be i.i.d. $\sim Q$. Consider the hypothesis testing between $H_1 : Q = P_1$ and $H_2 : Q = P_2$ and assume that $0 < D(P_1\|P_2), D(P_2\|P_1) < \infty$. Set $a = -D(P_2\|P_1)$, $\Delta = D(P_1\|P_2) + D(P_2\|P_1)$, and $\delta = D(P_1\|P_2) + \tau$ with $\tau \in (-D(P_1\|P_2), D(P_2\|P_1))$. Set also $r_i = \log\frac{p_1(y_i)}{p_2(y_i)}$ and*

$$\rho = \frac{1}{6n^2((D(P_1\|P_2) + D(P_2\|P_1))}.$$

Assume that BQ-CADMM runs sufficiently long such that either convergence or cycling occurs. Let the acceptance region for H_1 be

$$\mathcal{A}_n = \{y^n : \textit{BQ-CADMM cycles}\} \bigcup \{y^n : \textit{BQ-CADMM converges at } x^*_{\mathcal{Q}_\delta} = D(P_1\|P_2)\}.$$

Then we have

$$\liminf_{n\to\infty} -\frac{1}{n}\log\alpha_n = \Lambda^*(\tau),$$

and

$$\liminf_{n\to\infty} -\frac{1}{n}\log\beta_n = \Lambda^*(\tau) - \tau.$$

Proof. The δ-quantizer of this setup has its threshold at $-\tau$. The proof is similar to previous ones and we hence omit some details.

We first find a *necessary* condition for $y^n \in \mathcal{A}_n$. If $y^n \notin \mathcal{A}_n$, BQ-CADMM must reach a consensus at $x^*_{\mathcal{Q}_\delta} = -D(P_2\|P_1)$. Then Theorem 5.5 implies

$$\left|\mathcal{T}_{\mathcal{X}}\left(\frac{1}{n}\log\frac{p_1(y^n)}{p_2(y^n)}\right) + D(P_2\|P_1)\right| \le \left(1 + 4\rho\frac{m}{n}\right)\left(D(P_2\|P_1) - \tau\right).$$

Picking ρ small enough, e.g., $\rho < \frac{1}{4m(D(P_1\|P_2)-\tau)}$, we get

$$\frac{1}{n}\log\frac{p_1(y^n)}{p_2(y^n)} \le -\tau + \frac{1}{n}.$$

This further indicates

$$\mathcal{A}_n \subseteq \left\{y^n : \frac{1}{n}\log\frac{p_1(y^n)}{p_2(y^n)} > -\tau + \frac{1}{n}\right\}.$$

Therefore,

$$\begin{aligned}
&\liminf_{n\to\infty} -\frac{1}{n}\log\beta_n \\
\ge &\liminf_{n\to\infty} -\frac{1}{n}\log P_1\left(\left\{y^n : \frac{1}{n}\log\frac{p_1(y^n)}{p_2(y^n)} > -\tau + \frac{1}{n}\right\}\right) \\
= &\Lambda^*(\tau) - \tau, \qquad (5.16)
\end{aligned}$$

where the last equality is due to Lemma 5.2 and Theorem 5.4.

We next find a *sufficient* condition for $y^n \in \mathcal{A}_n$. If y^n results in a cyclic result of BQ-CADMM, we must have

$$\left|\frac{1}{n}\log\frac{p_1(y^n)}{p_2(y^n)} + \tau\right| \leq 6\rho n(D(P_1\|P_2) + D(P_2\|P_1)).$$

For the other case where the convergence is reached at $x^*_{\mathcal{Q}_\delta} = D(P_1\|P_2)$, we have

$$\left|\mathcal{T}_{\mathcal{X}}\left(\frac{1}{n}\log\frac{p_1(y^n)}{p_2(y^n)}\right) - D(P_1\|P_2)\right| \leq \left(1 + 4\rho\frac{m}{n}\right)(D(P_1\|P_2) + \tau).$$

With $\rho = \frac{1}{6n^2\Delta}$, we can verify that

$$\mathcal{A}_n \subseteq \left\{\frac{1}{n}\log\frac{p_1(y^n)}{p_2(y^n)} \geq -\tau - \frac{1}{n}\right\},$$

and thus

$$\liminf_{n\to\infty} -\frac{1}{n}\log\alpha_n \geq \Lambda^*(\tau). \tag{5.17}$$

Finally, the optimality of the Neyman-Pearson test and Theorem 5.4 establish the equalities in (5.16) and (5.17). □

Therefore, by replacing $\tau = \tau^*$ where τ^* is the optimal value in the centralized case, each node achieves the optimal centralized error exponent in Theorem 5.4 under the Neyman-Pearson criterion with exponential constraint.

5.3.4 Remarks

We have the following remarks regarding to our main results.

5.3.4.1 *Parameter Selection*

We focus on the Neyman-Pearson criterion with exponential constraint as both Stein's lemma and Chernoff theorem can be deduced from it. It is noted that we simply pick *one* setup for the δ-quantizer to accommodate all possible $-D(P_1\|P_2) < \tau < D(P_2\|P_1)$ in Theorem 5.9. Indeed, the quantizer parameters can be chosen from a quite broad set. With given τ in the Neyman-Pearson test of (5.1), one can pick any a and Δ to satisfy $a < -\tau < a + \Delta$. Then δ can be chosen such that $0 < \delta < \Delta$ and the threshold $a + \Delta - \delta \to -\tau$ as $n \to \infty$. To guarantee the optimal asymptotic performance, the algorithm parameter ρ must be small enough making $6\rho n\Delta \to 0$ with $n \to \infty$. This setup of quantizer and algorithm parameters can be similarly verified by exploring the sufficient and necessary conditions on the acceptance region.

It is worth noting that while the optimal error exponent only requires the constructed acceptance region to asymptotically approach the centralized one, suitable quantizer setup and step size can improve significantly the non-asymptotic performance; see Section 5.4 and Section 5.5.1. Besides, our choice of ρ is obtained from worst-case consensus error bounds that are generally loose and in turn result in loose ρ. Thus, our algorithm is very likely to perform well without requiring ρ to be very small; see also the simulation result in Section 5.5.1 when n is small.

5.3.4.2 *Practical Concerns*

In our main theorems, the acceptance regions all rely on the average LLR and seem to require knowledge of all the observations y^n. Fortunately, they can be fully characterized by local results according to Theorem 5.5. Assume that either convergence or cycling has been reached. Then Theorem 5.5 guarantees that a consensus is reached at all the nodes. That is, if a node converges at a (or $a + \Delta$), every other node reaches a convergence at a (or $a + \Delta$); if the node cycles, every other node cycles. As such, each node can make the same decision determined by the acceptance region.

Another concern might be on the stopping criterion at each node. A natural approach is to set the maximum number of iterations at the beginning, which then requires characterization on

convergence time of BQ-CADMM. In light of this, our present results lacks theoretic analysis on convergence time and we alternatively conduct simulations to see how fast BQ-CADMM proceeds. An upper bound on convergence time, which may depend on the network topology and agents' data, may still be insufficient as these quantities are locally unknown. There also exist some approaches that run additional algorithms to determine if a consensus has been reached in a fully distributed manner (see, e.g., [57, 58]). Noticing that these additional algorithms may take much time in large networks, we doubt if it is worth to use them due to the extra data communications. In general, we have not yet seen any efficient and guaranteed solution to stopping distributed averaging algorithms.

As to determining the cyclic state, nodes may record a certain number of consecutive variable values and check if any cycle exists. Indeed, the cyclic behavior can be ignored without losing any optimality in terms of error exponents. To see this, consider rejecting H_1 if oscillation happens. It can be similarly shown that the same error exponent is achieved under each criterion. Therefore, nodes can make their decision based on their current state to achieve optimal asymptotic performance, but they fail to reach a consensus when oscillation occurs. Moreover, as shown by our simulation in Section 5.5.1, convergence almost always happens with the choice of $\rho = \frac{1}{4m}$ when n becomes large.

5.3.4.3 Comparison with Fusion Center Based Structures

By enabling sensors and fusion center to transmit and receive data, the parallel and tandem networks are equivalent to the undirected star and path graphs, respectively. As such, the parallel and tandem networks can be regarded as special cases of consensus type structures. It is important to note that we achieve the optimal error exponent as in centralized settings at a cost of more data transmissions. The fusion center based structures need n and $n-1$ data transmissions for the parallel and tandem networks, respectively. The consensus based structure, however, has each sensor sending one bit to its neighbors and hence there are in total $2m$ bits per iteration. To see how many bits are needed for decision making at all nodes, it requires the characterization on the

convergence time of BQ-CADMM. While the current work cannot characterize convergence time, we will conduct numerical examples in Section 5.5 to evaluate it empirically.

5.4 Non-asymptotic Performance

While this chapter targets asymptotic characterizations on detection error probabilities, we are also interested in non-asymptotic performance of the proposed approach. For a broad class of criteria, including the Neyman-Pearson criterion and Bayesian criterion, the optimal acceptance region for H_1 is defined by a likelihood ratio test with a suitably chosen threshold

$$\mathcal{A}_n^* = \left\{ y^n : \frac{1}{n} \log \frac{p_1(y^n)}{p_2(y^n)} > \tau^* \right\},$$

where $\tau^* \in \mathbb{R}$ and n is finite. Similar to the asymptotic setting, we will use the consensus result of BQ-CADMM to construct an acceptance region whose type-I and type-II error probabilities are arbitrarily close to the centralized ones.

Set $a = \tau^* - 1$, $\Delta = 2$, and $\delta = 1$ for the δ-quantizer. Then the threshold in this setup is τ^*. Set also $r_i = \log \frac{p_1(y_i)}{p_2(y_i)}$. We again run BQ-CADMM long enough such that either a convergent result or cyclic result is reached. Let the acceptance region for H_1 be

$$\mathcal{A}_n = \{y^n : \text{BQ-CADMM cycles}\} \bigcup \{y^n : \text{BQ-CADMM converges at } x^*_{\mathcal{Q}_\delta} = \tau^* + 1\}.$$

We next find sufficient and necessary conditions for $y^n \in \mathcal{A}_n$. If $y^n \notin \mathcal{A}_n$, BQ-CADMM converges at $x^*_{\mathcal{Q}_\delta} = \tau^* - 1$. We get

$$|\tau^* - 1 - \mathcal{T}_{\mathcal{X}}(\bar{r})| \leq \left(1 + 4\rho \frac{m}{n}\right).$$

Then the projection operator $\mathcal{T}_{\mathcal{X}}(\cdot)$ implies that we must have $y^n \in \mathcal{A}_n$ if

$$\bar{r} - \tau^* > 4\rho\frac{m}{n}.$$

When $y^n \in \mathcal{A}_n$, BQ-CADMM either cycles or converges at $x^*_{\mathcal{Q}_\delta} = \tau^* + 1$. Thus, y^n must be such that either

$$|\bar{r} - \tau^*| < 12\rho n, \text{ or } \bar{r} - \tau^* \geq -4\rho\frac{m}{n}.$$

Thus, for $Q \in \{P_1, P_2\}$ we get

$$\begin{aligned}
&Q\left(\left\{y^n : \frac{1}{n}\log\frac{p_1(y^n)}{p_2(y^n)} > \tau^* + 4\rho\frac{m}{n}\right\}\right) \\
&\leq Q(\mathcal{A}_n) \\
&\leq Q\left(\left\{y^n : \frac{1}{n}\log\frac{p_1(y^n)}{p_2(y^n)} > \tau^* - \max\left\{12\rho n, 4\rho\frac{m}{n}\right\}\right\}\right).
\end{aligned}$$

Let $z = \frac{1}{n}\log\frac{p_1(y^n)}{p_2(y^n)}$ and denote its cumulative distribution function as $Q(\tau) = Q\left(\{z : z \leq \tau\}\right)$. We can further write the above as

$$1 - Q\left(\tau^* + 4\rho\frac{m}{n}\right) \leq Q(\mathcal{A}_n) \leq 1 - Q\left(\tau^* - \max\left\{12\rho n, 4\rho\frac{m}{n}\right\}\right).$$

Recall that $Q(\mathcal{A}_n^*) = 1 - Q(\tau^*)$, we have that $Q(\mathcal{A}_n) \to Q(\mathcal{A}_n^*)$ as $\rho \to 0$ given that $Q(\tau)$ is continuous at $\tau = \tau^*$. Therefore, when the continuity condition holds for both P_1 and P_2, we can make the type-I and type-II error probabilities arbitrarily close to the optimal ones by picking small enough ρ and attain the same centralized performance asymptotically under each criterion.

5.5 Simulations

5.5.1 Non-asymptotic Performance

We first consider the following hypothesis testing problem between two Gaussian distributions in star networks:

- $H_1 : P_1 = \mathcal{N}(1, 10)$,
- $H_2 : P_2 = \mathcal{N}(-1, 10)$.

For finite n and positive π_1 and π_2, the optimal centralized acceptance region for H_1 under the MAP criterion is

$$\left\{ y^n : \frac{1}{n}\left(2\sum_{i=1}^{n} y_i\right) > \frac{1}{n}\ln\frac{\pi_2}{\pi_1} \right\},$$

and the corresponding Bayesian error probability is given by

$$P_e = \pi_1 \mathrm{qfunc}\left(\left(1 - \frac{1}{2n}\ln\frac{\pi_2}{\pi_1}\right)\sqrt{\frac{n}{10}}\right) + \pi_2 \mathrm{qfunc}\left(\left(1 + \frac{1}{2n}\ln\frac{\pi_2}{\pi_1}\right)\sqrt{\frac{n}{10}}\right), \tag{5.18}$$

where $\mathrm{qfunc}(\cdot)$ denotes the complementary distribution function for standard Gaussian distribution. We perform Monte Carlo simulations to estimate the actual Bayesian error probability of our approach and compare it with (5.18). In the spirit of Remark 5.4, we run BQ-CADMM with $\rho = \frac{1}{4m}$ and check if BQ-CADMM converges to the right hypothesis. If BQ-CADMM cycles, we rerun BQ-CADMM with $\rho = \frac{1}{12n^2}$ and make the decision based on the new consensus result. Summarized in Fig. 5.1 and Fig. 5.2 are the Monte Carlo results for the Gaussian example with different prior probabilities: $\pi_1 = 0.5$ and $\pi_1 = 0.1$.

We observe from Fig. 5.1 that the consensus based error probabilities of the two cases are both very close to the centralized error probability with $\pi_1 = 0.5$ for all n. That said, in the case of $\pi_1 = 0.1$, the consensus based approach has its error probability far from the centralized one when n is small. This is because we use $\delta = 1$ in the consensus based approach for all positive prior

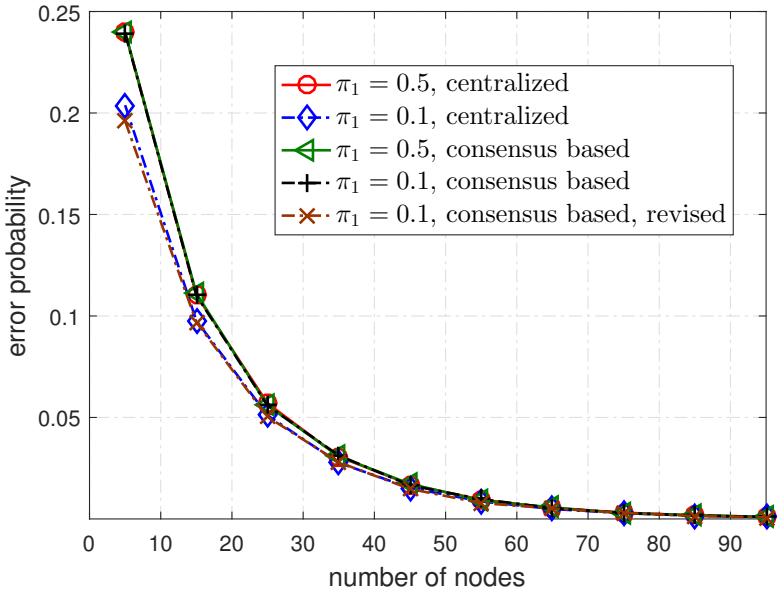


Fig. 5.1: Error probability of Monte Carlo simulations for the Gaussian example; the number of trials for each plotted value is 10^5.

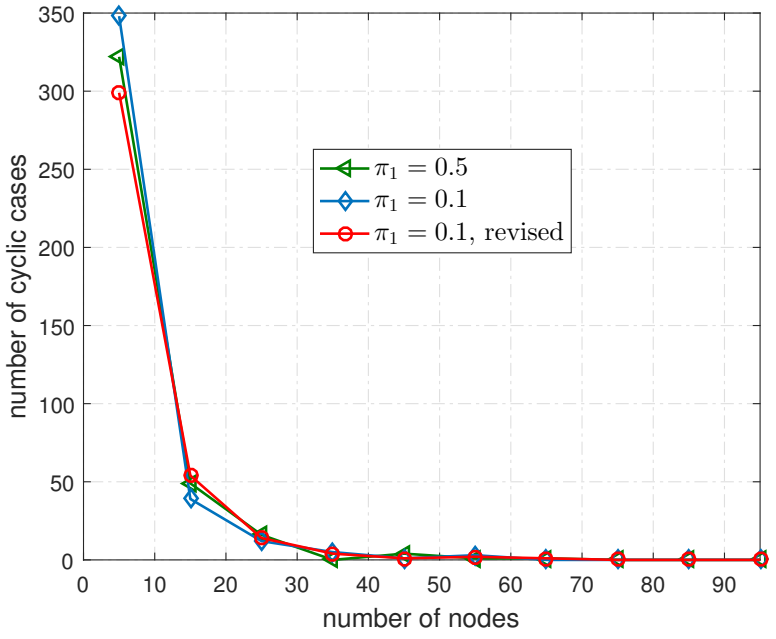


Fig. 5.2: Number of cyclic cases in 10^5 trials for the Gaussian example.

probabilities and consequently, the acceptance region is very different from the optimal centralized one with $\pi_1 = 0.1$ for small n. Similar to the setup of the δ-quantizer in Section 5.4, we pick $\delta = 1 + \frac{1}{n}\log\frac{\pi_2}{\pi_1}$ such that the threshold of the δ-quantizer becomes $\frac{1}{n}\log\frac{\pi_2}{\pi_1}$. Running the example with $\pi_1 = 0.1$ again, we obtain the new error probability very close to the optimal one for all n (see the dashed line with cross markers in Fig. 5.1). In addition, we record the number of trials in which BQ-CADMM cycles with $\rho = \frac{1}{4m}$. As plotted in Fig. 5.2, it is lcear that BQ-CADMM tends to converge as n increases.

5.5.2 Convergence Time

Our next simulation investigates the convergence time of the proposed approach. Given the number of nodes, we consider the above Gaussian example with $\pi_1 = 0.5$ over star graph which has the smallest number of edges for a connected network, complete graph which has the largest number of edges, and randomly generated connected graphs with intermediate numbers of edges. A random graph with n nodes and m edges is generated as follows: first generate a complete graph of n nodes and then randomly remove $\frac{n(n-1)}{2} - m$ edges while ensuring the graph stays connected. Since we have shown that BQ-CADMM with this example is more likely to converge with larger n, we only count the convergent cases and pick $\rho = \frac{1}{4m}$ for BQ-CADMM.

Simulation result is shown in Fig. 5.3. The plotted value is the average of $2,000$ runs in which both data and graph are randomly generated at each run. One can see that sparser and larger networks usually have longer convergence time and that the average convergence time for all cases is approximately $O(n\log n)$. As a result, in a sensor network with n nodes and m edges, the proposed approach requires approximately $O(mn\log n)$ bits of data transmissions for the Gaussian example.

Noticing that the above simulation uses a fixed algorithm parameter for BQ-CADMM, we now apply a decreasing parameter strategy which is shown to dramatically reduce the convergence time in Chapter 4. Start with $\rho = \frac{n}{m}$. If $\rho > \frac{1}{4m}$, we run BQ-CADMM for 50 iterations and then reduce it by a factor of 10. We repeat this process until $\rho \leq \frac{1}{4m}$ with which we run BQ-CADMM long

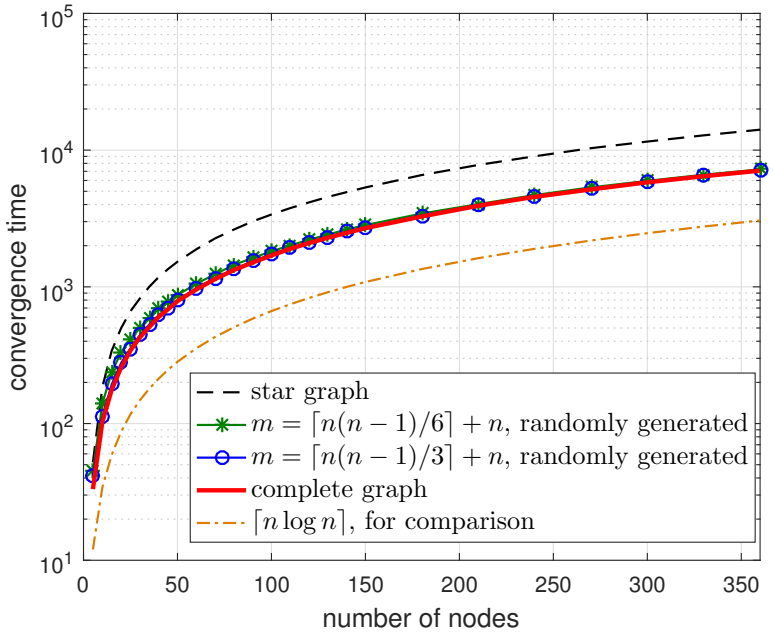


Fig. 5.3: Convergence time of BQ-CADMM for the Gaussian example with $\pi_1 = 0.5$; each plotted value is the average of $2,000$ runs.

enough such that either convergence or cycling occurs. The average convergence time is shown in Fig. 5.4. Comparing with fixed parameter strategy, we observe that the decreasing strategy runs $50\lceil \log_{10}(4n) \rceil$ iterations before ρ meets the accuracy requirement and makes BQ-CADMM proceed faster at early stages. When ρ indeed satisfies the required accuracy, it only takes a few iterations before reaching the final state. With the decreasing parameter strategy for BQ-CADMM, we conjecture that the consensus based approach requires $O(m \log n)$ bits of data transmissions on the average to achieve the optimal asymptotic performance for the Gaussian example.

5.6 Summary

This chapter studies asymptotic performance of consensus based detection using BQ-CADMM. Different from original BQ-CADMM that has a constant term in the consensus error bound, we construct a binary quantizer with controllable threshold such that the consensus value can be of desired accuracy. We then show that each node can achieve the same optimal error exponent as in

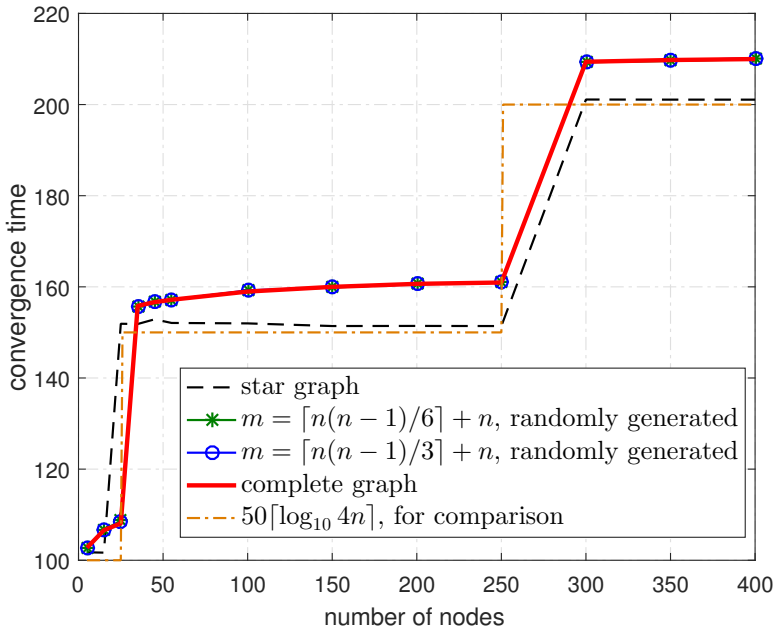


Fig. 5.4: Convergence time of BQ-CADMM using decreasing parameter strategy for the Gaussian example with $\pi_1 = 0.5$; each plotted value is the average of $2,000$ runs.

centralized cases under three common criteria. Non-asymptotic behavior of the proposed approach is also addressed.

5.7 Proof of Theorem 5.5

Proof. By the definition of $\mathcal{Q}_\delta(\cdot)$ in (5.2), it is clear that $|\mathcal{Q}_\delta(x) - \mathcal{T}_\mathcal{X}(x)| < \Delta < \infty$ for $x \in \mathbb{R}$. Together with the deterministic property (i.e., $\mathcal{Q}_\delta(x_1) = \mathcal{Q}_\delta(x_2)$ if $x_1 = x_2$), that BQ-CADMM using this δ-quantizer either converges or cycles with every node having the same sum of quantized variable values over one period can be shown by the same idea as that of Theorem 4.1. What remains is to derive the error bounds in the respective cases.

Convergent case: With convergence, we can write $x_i^k = x_i^*$ for $k \geq k_0$. To show the error bound (5.3), denote $e_i^* = \mathcal{T}_\mathcal{X}(x_i^*) - x_{\mathcal{Q}_\delta}^*$. Consider first $x_{\mathcal{Q}_\delta}^* = a$ and assume that $|x_{\mathcal{Q}_\delta}^* - \mathcal{T}_\mathcal{X}(\bar{r})| > \Delta - \delta$; otherwise, the error bound holds trivially. Following the same steps of the consensus error proof

of Theorem 4.1, we obtain that

$$\left|x^*_{\mathcal{Q}_\delta} - \mathcal{T}_{\mathcal{X}}(\bar{r})\right| \leq \frac{1}{n}\sum_{i=1}^{n}\left(2\rho|\mathcal{N}_i|+1\right)|e_i^*|.$$

Since $\sum_{i=1}^{n}|\mathcal{N}_i| = 2m$, it remains to find an upper bound for e_i^*. Note that $\mathcal{Q}_\delta(x_i^*) = x^*_{\mathcal{Q}_\delta} = a$ implies $\mathcal{T}_{\mathcal{X}}(x_i^*) \in [a, a+\Delta-\delta]$ and hence $|e_i^*| = |\mathcal{T}_{\mathcal{X}}(x_i^*) - a| \leq \Delta - \delta$. Similar argument proves the error bound for $x^*_{\mathcal{Q}_\delta} = a + \Delta$ and is omitted.

Cyclic case: We will show that x_i^k is close to the threshold $a + \Delta - \delta$ and then use this fact to characterize the difference between the threshold and the data average. Assume that the cycling state has been reached, i.e., $k \geq k_0$. With only two quantization values a and $a + \Delta$, there are at most four possible cases for two consecutive agent values $\mathcal{Q}_\delta(x_i^k)$ and $\mathcal{Q}_\delta(x_i^{k+1})$. We now discuss these cases one by one and before this, we write the following useful update from the x_i- and α_i-updates in Algorithm 1:

$$\begin{aligned}
x_i^{k+1} &= \frac{1}{1+2\rho|\mathcal{N}_i|}\left(\rho|\mathcal{N}_i|\mathcal{Q}_\delta(x_i^k) + \rho\sum_{j\in\mathcal{N}_i}\mathcal{Q}_\delta(x_j^k) - \alpha_i^k + r_i\right)\\
&= \frac{1}{1+2\rho|\mathcal{N}_i|}\left(\rho|\mathcal{N}_i|\mathcal{Q}_\delta(x_i^k) + \rho\sum_{j\in\mathcal{N}_i}\mathcal{Q}_\delta(x_j^k) - \alpha_i^{k-1} - \rho|\mathcal{N}_i|\mathcal{Q}_\delta(x_i^k) + \rho\sum_{j\in\mathcal{N}_i}\mathcal{Q}_\delta(x_j^k) + r_i\right)\\
&= x_i^k + \frac{1}{1+2\rho|\mathcal{N}_i|}\left(2\rho\sum_{j\in\mathcal{N}_i}\mathcal{Q}_\delta(x_j^k) - \rho|\mathcal{N}_i|\mathcal{Q}_\delta(x_i^{k-1}) - \rho\sum_{j\in\mathcal{N}_i}\mathcal{Q}_\delta(x_j^{k-1})\right). \qquad (5.19)
\end{aligned}$$

- Case 1: $\mathcal{Q}_\delta(x_i^k) = a$ and $\mathcal{Q}_\delta(x_i^{k+1}) = a + \Delta$. By the definition of δ-quantizer, we have $x_i^k \leq a + \Delta - \delta < x_i^{k+1}$. Following (5.19) and using the fact that only a and $a + \Delta$ can be the output of $\mathcal{Q}_\delta(\cdot)$, we also get $x_i^{k+1} \leq a + \Delta - \delta + \frac{2\rho|\mathcal{N}_i|}{1+2\rho|\mathcal{N}_i|}\Delta$. In summary, we have

$$a + \Delta - \delta < x_i^{k+1} \leq a + \Delta - \delta + \frac{2\rho|\mathcal{N}_i|}{1+2\rho|\mathcal{N}_i|}\Delta.$$

- Case 2: $\mathcal{Q}_\delta(x_i^k) = a + \Delta$ and $\mathcal{Q}_\delta(x_i^{k+1}) = a$. Similar to Case 1, it can be shown that

$$a + \Delta - \delta - \frac{2\rho|\mathcal{N}_i|}{1 + 2\rho|\mathcal{N}_i|}\Delta < x_i^{k+1} \leq a + \Delta - \delta.$$

- Case 3: $\mathcal{Q}_\delta(x_i^k) = a + \Delta$ and $\mathcal{Q}_\delta(x_i^{k+1}) = a + \Delta$. We can immediately conclude that $x_i^{k+1} > a + \Delta - \delta$. To find an upper bound on x_i^{k+1}, consider the α_i-update at index k:

$$\alpha_i^k = \alpha_i^{k-1} + \rho|\mathcal{N}_i|\mathcal{Q}_\delta(x_i^k) - \rho \sum_{j \in \mathcal{N}_i} \mathcal{Q}_\delta(x_j^k) \geq \alpha_i^{k-1},$$

where inequality follows from $\mathcal{Q}_\delta(x_i^k) = a + \Delta$. By induction, we have $\alpha_i^{k'} \leq \alpha_i^k$ where $k' < k$ is the largest index such that $\mathcal{Q}_\delta(x_i^{k'}) = a$. Note that such k' always exists for $k \geq k_0 + T$ as a result of the cyclic behavior and (5.4). Then we have

$$\begin{aligned}
x_i^{k+1} &= \frac{1}{1 + 2\rho|\mathcal{N}_i|}\left(\rho|\mathcal{N}_i|\mathcal{Q}_\delta(x_i^k) + \rho \sum_{j \in \mathcal{N}_i} \mathcal{Q}_\delta(x_j^k) - \alpha_i^k + r_i\right) \\
&\leq \frac{1}{1 + 2\rho|\mathcal{N}_i|}\left(\rho|\mathcal{N}_i|\mathcal{Q}_\delta(x_i^k) + \rho \sum_{j \in \mathcal{N}_i} \mathcal{Q}_\delta(x_j^k) - \alpha_i^{k'} + r_i\right) \\
&= x_i^{k'} + \frac{1}{1 + 2\rho|\mathcal{N}_i|}\Bigg(\rho|\mathcal{N}_i|\mathcal{Q}_\delta(x_i^k) + \rho \sum_{j \in \mathcal{N}_i} \mathcal{Q}_\delta(x_j^k) + \rho \sum_{j \in \mathcal{N}_i} \mathcal{Q}_\delta(x_j^{k'}) - \rho|\mathcal{N}_i|\mathcal{Q}_\delta(x_i^{k'}) \\
&\quad - \rho|\mathcal{N}_i|\mathcal{Q}_\delta(x_i^{k'-1}) - \rho \sum_{j \in \mathcal{N}_i} \mathcal{Q}_\delta(x_j^{k'-1})\Bigg),
\end{aligned} \tag{5.20}$$

where the last equality is obtained by the x_i- and α_i-updates at the k'-th iteration. As $\mathcal{Q}_\delta(x_i^{k'}) = a$, we have

$$x_i^{k+1} \leq a + \Delta - \delta + \frac{3\rho|\mathcal{N}_i|}{1 + 2\rho|\mathcal{N}_i|}\Delta.$$

- Case 4: $\mathcal{Q}_\delta(x_i^k) = a$ and $\mathcal{Q}_\delta(x_i^{k+1}) = a$. Similar to Case 3, we can get

$$a + \Delta - \delta - \frac{3\rho|\mathcal{N}_i|}{1 + 2\rho|\mathcal{N}_i|}\Delta < x_i^{k+1} \leq a + \Delta - \delta.$$

Summarizing all the four cases and using the fact that $|\mathcal{N}_i| < n$, we conclude the following:

$$|x_i^k - (a + \Delta - \delta)| \leq \frac{3\rho|\mathcal{N}_i|}{1 + 2\rho|\mathcal{N}_i|}\Delta < \frac{3\rho n}{1 + 2\rho n}\Delta, \text{ for } k \geq k_0. \tag{5.21}$$

With (5.21), we can now bound $x_i^k - \mathcal{Q}_\delta(x_i^k)$ with $k \geq k_0$. If $\mathcal{Q}_\delta(x_i^k) = a$, then

$$a + \Delta - \delta - \frac{3\rho n}{1 + 2\rho n}\Delta < x_i^k \leq a + \Delta - \delta.$$

Thus,

$$|x_i^k - \mathcal{Q}_\delta(x_i^k)| \leq \max\left\{\left|\Delta - \delta - \frac{3\rho n}{1 + 2\rho n}\Delta\right|, \Delta - \delta\right\}.$$

When $\mathcal{Q}_\delta(x_i^k) = a + \Delta$, we can similarly derive

$$|x_i^k - \mathcal{Q}_\delta(x_i^k)| \leq \max\left\{\left|\frac{3\rho n}{1 + 2\rho n}\Delta - \delta\right|, \delta\right\}.$$

As $0 < \delta < \Delta$, we finally have

$$|x_i^k - \mathcal{Q}_\delta(x_i^k)| < \frac{3}{2}\Delta. \tag{5.22}$$

To bound the difference between the threshold $a + \Delta - \delta$ and the data average $\bar{r}$, we sum up the variable values over one period for $k \geq k_0$ and get

$$\left|\frac{\sum_{l=0}^{T-1} x_i^{k+l}}{T} - (a + \Delta - \delta)\right| < \frac{3\rho n}{1 + 2\rho n}\Delta, \tag{5.23}$$

and

$$\left| \frac{\sum_{l=0}^{T-1} x_i^{k+l}}{T} - \frac{\sum_{l=0}^{T-1} \mathcal{Q}_\delta(x_i^{k+l})}{T} \right| < \frac{3}{2}\Delta, \tag{5.24}$$

where (5.23) is from (5.21) and (5.24) is from (5.22). Also summing up both sides of the x_i-update of BQ-CADMM over a period yields that

$$\begin{aligned}(1+2\rho|\mathcal{N}_i|)\frac{\sum_{l=0}^{T-1} x_i^{k+l}}{T} - \rho|\mathcal{N}_i\frac{\sum_{l=0}^{T-1} \mathcal{Q}_\delta(x_i^{k+l})}{T} \\ - \rho \sum_{j\in\mathcal{N}_i} \frac{\sum_{l=0}^{T-1} \mathcal{Q}_\delta(x_j^{k+l})}{T} + \frac{\sum_{l=0}^{T-1} \alpha_i^{k+l}}{T} - r_i = 0.\end{aligned} \tag{5.25}$$

Further summing up both sides (5.25) from $i = 1$ to n, we have

$$\sum_{i=1}^{n}(1+2\rho|\mathcal{N}_i|)\frac{\sum_{l=0}^{T-1} x_i^{k+l}}{T} - \sum_{i=1}^{n} 2\rho|\mathcal{N}_i|\frac{\sum_{l=0}^{T-1} \mathcal{Q}_\delta(x_i^{k+l})}{T} - \sum_{i=1}^{n} r_i = 0, \tag{5.26}$$

where we use the fact $\sum_{i=1}^{n} \alpha_i^k = 0$ for any k (cf. Section 4.7) together with (5.4). To complete the proof, we divide both sides of (5.26) by n and use (5.23) and (5.24), which leads to

$$|\bar{r} - (a + \Delta - \delta)| < 4\rho\frac{m}{n}\frac{3}{2}\Delta + \frac{3\rho n}{1+2\rho n} < 6\rho n\Delta.$$

where the second inequality is because $m \leq \frac{n(n-1)}{2}$ for a connected undirected graph. □

CHAPTER 6
CONCLUSION AND FUTURE WORK

This dissertation develops efficient ADMM based algorithms for quantized consensus in large scale networks. With dithered quantization, PQ-CADMM achieves asymptotic convergence to the exact average. When deterministic quantization is employed, the modified CADMM algorithms, DQ-CADMM and BQ-CADMM, either converge to a consensus or cycle with the sample mean over a period reaching a consensus. We also establish consensus error bounds that do not depend on agents data nor the network size. The effect of ADMM step size is studied and a decreasing strategy is proposed to accelerate the algorithm under certain consensus accuracy guarantee. Finally, we apply BQ-CADMM to distributed detection in connected sensor networks where sensors can only communicate with their immediate neighbors at each iteration. By employing an identical one-bit quantizer, each node can achieve the optimal asymptotic performance of centralized detection under three commonly used criteria.

The current results also motivate several future research directions:

6.1 Characterization on Convergence Time

Our simulations show that DQ-CADMM and BQ-CADMM with fixed step size proceed very fast to the convergent value or the cyclic period. Indeed, the decreasing strategy for step size leads

to approximately $O(\log n)$ convergence time in the Gaussian example in Section 5.5.2. However, we still lack the characterizations on convergence time and cyclic period, which play a key role in implementing the algorithms, e.g., setting the maximum iteration number for DQ-CADMM or BQ-CADMM to guarantee convergent or cyclic states. The difficult part for the present approach is that only the deterministic and bounded error properties of deterministic quantization schemes can be utilized.

6.2 General Convex Functions

In this dissertation, we only use CADMM to solve distributed average consensus by formulating it as a least-squares problem. The CADMM algorithm, in fact, can handle general convex objective functions. Thus, it is interesting to investigate how a quantized version of the CADMM works for general convex objectives. This consideration is hard since the local functions, unlike quadratic functions that have linear gradients, may have nonlinear gradient or even be non-smooth. Some initial results have been reported in [59], yet further effort is needed to broaden the results to more general cases.

6.3 Online/Sequential Setting for Distributed Detection

In Chapter 5, we consider asymptotic characterizations with respect to the number of sensors, while each sensor has only one observation. Another popular setting is that the number of sensors is finite and that each sensor may consecutively receive observations about the common phenomenon. To our best knowledge, there are no results characterizing the asymptotic performance when one-bit data communication is used. It is possible to extend our approach to these setting by accommodating temporal observations.

REFERENCES

[1] J. N. Tsitsiklis, "Problems in decentralized decision making and computation," DTIC Document, Tech. Rep., 1984.

[2] J. Tsitsiklis, D. Bertsekas, and M. Athans, "Distributed asynchronous deterministic and stochastic gradient optimization algorithms," *IEEE Trans. Autom. Control*, vol. 31, no. 9, pp. 803–812, Sep. 1986.

[3] T. Vicsek, A. Czirók, E. Ben-Jacob, I. Cohen, and O. Shochet, "Novel type of phase transition in a system of self-driven particles," *Physical Review Letters*, vol. 75, no. 6, pp. 1226–1229, 1995.

[4] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Transactions on automatic control*, vol. 48, no. 6, pp. 988–1001, 2003.

[5] W. Ren and R. W. Beard, "Consensus seeking in multiagent systems under dynamically changing interaction topologies," *IEEE Trans. Autom. Control*, vol. 50, no. 5, pp. 655–661, May 2005.

[6] S. Kar, R. Tandon, H. V. Poor, and S. Cui, "Distributed detection in noisy sensor networks," in *Proc. IEEE Int. Symp. Inf. Theory*, Saint Petersburg, Russia, Jul. 2011.

[7] S. Kar and J. M. F. Moura, "Consensus + innovations distributed inference over networks: cooperation and sensing in networked systems," *IEEE Signal. Proc. Mag.*, vol. 30, no. 3, pp. 99–109, May 2013.

[8] C. Xu and F. C. Lau, *Load Balancing in Parallel Computers: Theory and Practice*. Dordrecht, Germany: Kluwer, 1997.

[9] A. Kashyap, T. Başar, and R. Srikant, "Quantized consensus," *Automatica*, vol. 43, no. 7, pp. 1192–1203, 2007.

[10] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Syst. Contr. Lett.*, vol. 53, no. 1, pp. 65–78, 2004.

[11] D. Jakovetić, J. Xavier, and J. M. F. Moura, "Weight optimization for consensus algorithms with correlated switching topology," *IEEE Trans. Signal Process.*, vol. 58, no. 7, pp. 3788–3801, Jul. 2010.

[12] A. Nedic, A. Olshevsky, A. Ozdaglar, and J. N. Tsitsiklis, "On distributed averaging algorithms and quantization effects," *IEEE Trans. Autom. Control*, vol. 54, no. 11, pp. 2506–2517, Nov. 2009.

[13] T. C. Aysal, M. E. Yildiz, A. D. Sarwate, and A. Scaglione, "Broadcast gossip algorithms for consensus," *IEEE Trans. Signal Process.*, vol. 57, no. 7, pp. 2748–2761, Jul. 2009.

[14] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Randomized gossip algorithms," *IEEE Trans. Inf. Theory*, vol. 52, no. 6, pp. 2508–2530, Jun. 2006.

[15] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.

[16] I. D. Schizas, A. Ribeiro, and G. B. Giannakis, "Consensus in ad hoc WSNs with noisy links—Part I: Distributed estimation of deterministic signals," *IEEE Trans. Signal Process.*, vol. 56, no. 1, pp. 350–364, Jan. 2008.

[17] H. Zhu, G. B. Giannakis, and A. Cano, "Distributed in-network channel decoding," *IEEE Trans Signal Process*, vol. 57, no. 10, pp. 3970–3983, Oct. 2009.

[18] T. Erseghe, D. Zennaro, E. Dall'Anese, and L. Vangelista, "Fast consensus by the alternating direction multipliers method," *IEEE Trans. Signal Process.*, vol. 59, no. 11, pp. 5523–5537, Nov. 2011.

[19] L. Schuchman, "Dither signals and their effect on quantization noise," *IEEE T. Commun. Techn.*, vol. 12, no. 4, pp. 162–165, Dec. 1964.

[20] T. C. Aysal, M. J. Coates, and M. G. Rabbat, "Distributed average consensus with dithered quantization," *IEEE Trans. Signal Process.*, vol. 56, no. 10, pp. 4905–4918, Oct. 2008.

[21] S. Kar and J. M. Moura, "Distributed consensus algorithms in sensor networks: quantized data and random link failures," *IEEE Trans. Signal Process.*, vol. 58, no. 3, pp. 1383–1400, 2010.

[22] R. Carli, F. Fagnani, P. Frasca, and S. Zampieri, "Gossip consensus algorithms via quantized communication," *Automatica*, vol. 46, no. 1, pp. 70–80, 2010.

[23] T. Li, M. Fu, L. Xie, and J. F. Zhang, "Distributed consensus with limited communication data rate," *IEEE Trans. Autom. Control*, vol. 56, no. 2, pp. 279–292, Feb. 2011.

[24] M. E. Chamie, J. Liu, and T. Başar, "Design and analysis of distributed averaging with quantized communication," in *Proc. 53rd IEEE Conf. Decision and Control*, Los Angeles, CA, Dec. 2014.

[25] P. K. Varshney, *Distributed Detection and Data Fusion*. New York: Springer, 1997.

[26] R. Viswanathan and P. K. Varshney, "Distributed detection with multiple sensors: Part I—Fundamentals," *Proc. IEEE*, vol. 85, no. 1, pp. 54–63, 1997.

[27] 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, 1997.

[28] J. F. Chamberland and V. V. Veeravalli, "Decentralized detection in sensor networks," *IEEE Trans. Signal Process.*, vol. 51, no. 2, pp. 407–416, 2003.

[29] J. F. Chamberland and V. V. Veeravalli, "Asymptotic results for decentralized detection in power constrained wireless sensor networks," *IEEE J. Sel. Areas Commun.*, vol. 22, no. 6, pp. 1007–1015, Aug. 2004.

[30] J. N. Tsitsiklis, "Decentralized detection by a large number of sensors," *Mathematics of Control, Signals and Systems*, vol. 1, no. 2, pp. 167–182, 1988.

[31] W. P. Tay, J. N. Tsitsiklis, and M. Z. Win, "On the subexponential decay of detection error probabilities in long tandems," *IEEE Trans. Inf. Theory*, vol. 54, no. 10, pp. 4767–4771, 2008.

[32] V. Saligrama, M. Alanyali, and O. Savas, "Distributed detection in sensor networks with packet losses and finite capacity links," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4118–4132, Nov. 2006.

[33] S. Kar, S. Aldosari, and J. M. F. Moura, "Topology for distributed inference on graphs," *IEEE Trans. Signal Process.*, vol. 56, no. 6, pp. 2609–2613, Jun. 2008.

[34] D. Jakovetic, J. M. F. Moura, and J. Xavier, "Distributed detection over noisy networks: Large deviations analysis," *IEEE Trans. Signal Process.*, vol. 60, no. 8, pp. 4306–4320, Aug. 2012.

[35] D. Bajovic, D. Jakovetic, J. M. F. Moura, J. Xavier, and B. Sinopoli, "Large deviations performance of consensus+innovations distributed detection with non-gaussian observations," *IEEE Trans. Signal Process.*, vol. 60, no. 11, pp. 5987–6002, Nov. 2012.

[36] P. Braca, S. Marano, V. Matta, and P. Willett, "Asymptotic optimality of running consensus in testing binary hypotheses," *IEEE Trans. Signal Process.*, vol. 58, no. 2, pp. 814–825, Feb. 2010.

[37] B. He and X. Yuan, "On the $O(1/n)$ convergence rate of the Douglas-Rachford alternating direction method," *SIAM J. Numer. Anal.*, vol. 50, no. 2, pp. 700–709, 2012.

[38] M. Hong and Z.-Q. Luo, "On the linear convergence of the alternating direction method of multipliers," *arXiv preprint arXiv:1208.3922*, 2012.

[39] W. Deng and W. Yin, "On the global and linear convergence of the generalized alternating direction method of multipliers," *J. Sci. Comput.*, vol. 66, no. 3, pp. 889–916, 2016.

[40] S. Zhu and B. Chen, "Distributed average consensus with deterministic quantization: an ADMM approach," in *Proc. IEEE Global Conf. Signal and Information Processing (GlobalSIP)*, Orlando, FL, Dec. 2015.

[41] ——, "Quantized cosensus by the ADMM: probabilistic versus deterministic quantizers," *IEEE Trans. Signal Process.*, vol. 64, no. 7, pp. 1700–1713, Apr. 2016.

[42] ——, "Corrections to "quantized consensus by the ADMM: probabilistic versus deterministic quantizers"," *IEEE Trans. Signal Processing*, vol. 65, no. 6, pp. 1638–1639, 2017.

[43] ——, "Distributed average consensus with bounded quantization," in *Proc. IEEE Int. Workshop on Signal Process. Advances in Wireless Commun.*, Edinburgh, UK, Jul. 2016.

[44] ——, "Distributed average consensus with bounded quantizer and unbounded input," *arXiv preprint arXiv:1602.04193*, 2016.

[45] ——, "Distributed detection over connected networks via one-bit quantizer," in *Proc. IEEE Int. Symp. Inf. Theory*, Barcelona, Spain, Jul. 2016.

[46] ——, "Distributed detection in ad hoc networks through quantized consensus," *arXiv preprint arXiv 1612.01904*, 2016.

[47] F. R. Chung, *Spectral Graph Theory*. American Mathematical Soc., 1997, vol. 92.

[48] M. Fiedler, "Algebraic connectivity of graphs," *Czechoslovak Mathematical Journal*, vol. 23, no. 2, pp. 298–305, 1973.

[49] N. Parikh, S. Boyd *et al.*, "Proximal algorithms," *Foundations and Trends® in Optimization*, vol. 1, no. 3, pp. 127–239, 2014.

[50] W. Shi, Q. Ling, K. Yuan, G. Wu, and W. Yin, "On the linear convergence of the ADMM in decentralized consensus optimization," *IEEE Trans. Signal Process.*, vol. 62, no. 7, pp. 1750–1761, Apr. 2014.

[51] J.-J. Xiao and Z.-Q. Luo, "Decentralized estimation in an inhomogeneous sensing environment," *IEEE Trans. Inf. Theory*, vol. 51, no. 10, pp. 3564–3575, Oct. 2005.

[52] L. Xiao, S. Boyd, and S. Lall, "A scheme for robust distributed sensor fusion based on average consensus," in *Proc. Int. Symp. Information Processing in Sensor Networks*, Los Angeles, CA, Apr. 2005.

[53] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge Univ. Press, 2004.

[54] A. Dembo and O. Zeitouni, *Large Deviations Techniques and Applications*. New York: Springer, 2009.

[55] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed. New York: Wiley, 2006.

[56] W. Hoeffding, "Asymptotically optimal tests for multinomial distributions," *The Annals of Mathematical Statistics*, pp. 369–401, 1965.

[57] N. E. Manitara and C. N. Hadjicostis, "Distributed stopping for average consensus in undirected graphs via event-triggered strategies," *Automatica*, vol. 70, pp. 121–127, 2016.

[58] V. Yadav and M. V. Salapaka, "Distributed protocol for determining when averaging consensus is reached," in *45th Annual Allerton Conference on Communication, Control, and Computing*, Monticello, IL, 2007.

[59] S. Zhu, M. Hong, and B. Chen, "Quantized consensus ADMM for multi-agent distributed optimization," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Shanghai, China, March 2016.

VITA

NAME: Shengyu Zhu

EDUCATION

- Ph.D. in Electrical and Computer Engineering, Syracuse University, January 2012 – August 2017 (expected)
- M.S. in Mathematics, Syracuse University, January 2015 – August 2016
- B.E. in Electrical Engineering, Beijing Institute of Technology, September 2006 – July 2010

PUBLICATIONS

1. S. Zhu and B. Chen, “Distributed detection in ad hoc networks through quantized," *IEEE Transactions on Information Theory*, under review.
2. S. Zhu and B. Chen, “Distributed average consensus with bounded quantizer and unbounded input," *IEEE Transactions on Signal Processing*, under review.
3. S. Zhu and B. Chen, “Quantized consensus by the ADMM: probabilistic versus deterministic quantizers," *IEEE Transactions on Signal Processing*, vol. 64, no. 7, pp. 1700–1713, Apr. 2016.
4. G. Xu, S. Zhu, and B. Chen,“Decentralized data reduction with quantization constraints,” *IEEE Transactions on Signal Processing*, vol. 62, no. 7, pp. 1775–1784, Apr. 2014.
5. S. Zhu and B. Chen, “Distributed average consensus with bounded quantization,” in *Proc. IEEE International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*,

Edinburgh, UK, Jul. 2016.

6. S. Zhu and B. Chen, “Distributed detection over connected networks via one-bit quantizer,” in *Proc. IEEE International Symposium on Information Theory (ISIT)*, Barcelona, Spain, Jul. 2016.

7. S. Zhu, M. Hong, and B. Chen, “Quantized consensus ADMM for multi-agent distributed optimization," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Shanghai, China, Mar. 2016.

8. S. Zhu and B. Chen, “Distributed average consensus with deterministic quantization: an ADMM approach," in *Proc. IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, Orlando, FL, Dec. 2015.

9. S. Zhu, G. Xu, and B. Chen, “Are global sufficient statistics always sufficient: the impact of quantization on decentralized data reduction," in *Proc. Asilomar Conference on Signals, Systems, and Computers (Asilomar)*, Monterey, CA, Nov. 2013.

10. S. Zhu and B. Chen, “Data reduction in tandem fusion systems," in *Proc. IEEE China Summit and International Conference on Signal and Information Processing (ChinaSIP)*, Beijing, China, Jul. 2013.

11. S. Zhu, E. Akofor, and B. Chen, “Interactive distributed detection with conditionally independent observations," in *Proc. IEEE Wireless Communications and Networking Conference (WCNC)*, Shanghai, China, Apr. 2013.