2011

Distributed Synchronization and Spectrum Sensing in Cognitive Radio Networks

Gang Xiong
Lehigh University

Distributed Synchronization and Spectrum Sensing in Cognitive Radio Networks

by

Gang Xiong

Presented to the Graduate and Research Committee
of Lehigh University
in Candidacy for the Degree of
Doctor of Philosophy

in

Electrical Engineering
Lehigh University

May 2011

Approved and recommended for acceptance as a dissertation in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

Date

Dissertation Advisor

Accepted Date

Committee Members:

Shalinee Kishore

Tiffany Jing Li

Parv Venkitasubramaniam

Aylin Yener

Acknowledgements

I would like to express my deepest gratitude to my advisor Professor Shalinee Kishore for her support, patience and encouragement in all the time of my research and writing of this dissertation. I appreciate all her contributions of time, ideas, and funding to make my Ph.D. experience productive and stimulating. The open attitude of Professor Kishore towards research has granted me a vast freedom to explore my interest on my own. Moreover, I also deeply appreciate her invaluable support for me to spend more time with my wife.

I would like to thank Professor Tiffany Jing Li and Professor Parv Venkitasubrama at Lehigh University, and Professor Aylin Yener at the Pennsylvania State University for kindly serving on my PhD committees and providing many thorough suggestions and comments. I have benefited enormously from my discussions with them during my PhD study and in my general examination. In particular, I would like to thank Professor Yener for her insightful advice and critical review that improve my work remarkably, and for her invaluable support when I visited WCAN at the Pennsylvania State University in 2010.

I am grateful to my colleagues and friends at Lehigh university who have offered help in various ways. I thank my fellow graduate students in the group: Yan, Zhenlei, Yingda, Chen, Nanda, Andrew and Sarah. Special thanks to my former labmate in the Ohio State University, Yan for introducing me the precious opportunity to start my Ph.D study at Lehigh.

Last, but certainly not the least, I am deeply indebted to my wife, Xia, who has been with me all the time and supported me all way long, and my parents Xiaochun and Yuqing, my sister Ying for their constant love, support and encouragement. Without them, this can not be achieved.

Dedicated to my wife, my parents and my sister.

Contents

List of Figures

Abstract

As an emerging and promising technology, cognitive radio has been recently proposed to alleviate spectrum scarcity by allowing unlicensed (*secondary*) users to coexist with licensed (*primary*) users while not causing harmful interference. In this work, we study two important components in constructing cognitive radio networks: distributed time synchronization and cooperative spectrum sensing.

First, we focus on the task of synchronizing distributed cognitive radios to the same timing reference, so that they may effectively communicate over a common control channel and conduct network tasks, e.g., cooperative spectrum sensing, distributed spectrum allocation, etc.. Although presented here in the context of cognitive radio network formation, distributed timing synchronization is critical in all distributed network scenarios. In this dissertation, we propose a novel discrete time *second*- and *high-order* distributed consensus time synchronization (DCTS) algorithm for ad hoc networks and examine their convergence properties. We claim that the optimal convergence rate of the second- and high-order DCTS algorithm is superior to that of the first-order DCTS algorithm under an appropriate algorithm design. Furthermore, we extend our study on the convergence of the DCTS algorithm when both deterministic and uncertain time delays impact local pair-wise time information exchange. Specifically, we model random delay between secondary users using a Gaussian approximation and determine the resulting asymptotic behavior of global synchronization error.

In the second topic, we study cooperative spectrum sensing in cost constrained cognitive radio networks with a centralized fusion center. Specifically, we examine the case when cognitive radios forward local spectrum statistic to the fusion cen-

ter over two channel scenarios: parallel access channel (PAC) and multiple access channel (MAC). For both channel scenarios, we aim to maximize the global detection probability of available spectrum subject to a *system level* cost constraint. (1) In PAC scenario, our objective is to choose appropriate number of energy samples that must be collected at each secondary user and appropriate amplifier gain that each secondary user must use to forward its statistics to the fusion center. When jointly designing these two parameters, we demonstrate that only *one* secondary user needs to be active, i.e., collecting local energy samples and transmitting energy statistic to fusion center. (2) In MAC scenario, our objective is to choose appropriate beamforming weights subject to a global transmit power constraint. Under correlated lognormal shadowing, we derive closed-form expressions of optimal beamforming weights and claim that global detection probability increases as the number of secondary users increases for a simplified linear array network.

Chapter 1

Introduction

1.1 Background

1.1.1 Cognitive Radio

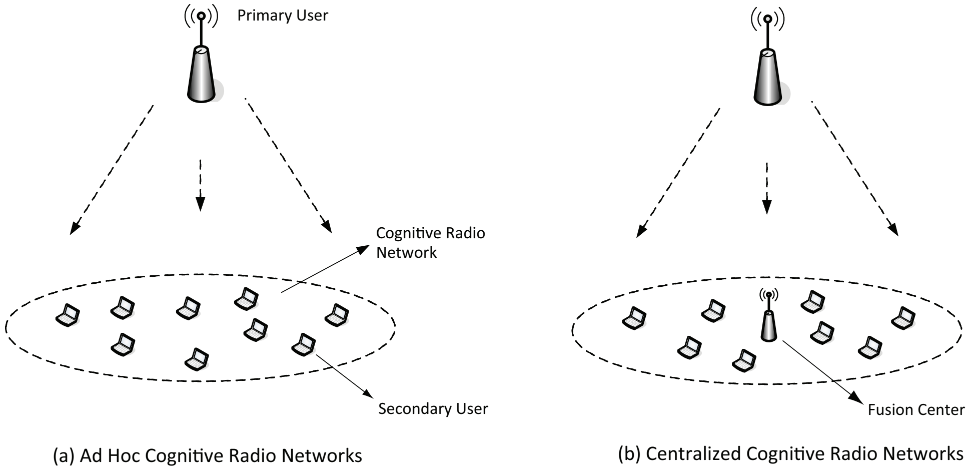
In 2002, spectrum policy task force organized by Federal Communications Commission (FCC) initialized a comprehensive review of spectrum policy. Preliminary spectrum measurements indicated that the assigned spectrum are severely underutilized in both spatial and temporal domains [1]. According to the report by FCC, some frequency bands are largely unoccupied most of the time; some other frequency bands are only partially occupied and the remaining frequency bands are heavily used [2, 3]. In order to enhance spectrum utility and alleviate spectrum scarcity, cognitive radio has been recently proposed as a potential solution. A cognitive radio (*secondary user*) has the adaptability and agility to coexist with primary users in a licensed band without causing harmful interference. One of ongoing standards based on cognitive radio technology is IEEE 802.22 wireless regional area networks (WRAN), which aims to provide broadband access services in spectrum allocated to TV broadcast service [4]. It has been shown that these spectrum bands are especially underutilized in rural areas and may be used for (secondary) wireless broadband services.

The term and concept of *cognitive radio* were first introduced by Mitola [5], where radio knowledge representation language (RKRL) was described as a underlying language for cognitive radio to enhance the flexibility of personal wireless services. Later, a formal definition of cognitive radio was given by Haykin [3]:

> *"Cognitive radio is an intelligent wireless communication system that is aware of its surrounding environment (i.e., outside world), and uses the methodology of understanding-by-building to learn from the environment and adapt its internal states to statistical variations in the incoming RF stimuli by making corresponding changes in certain operating parameters (e.g., transmit-power, carrier-frequency, and modulation strategy) in real-time, with two primary objectives in mind: 1) highly reliable communications whenever and wherever needed; 2) efficient utilization of the radio spectrum."*

In other words, a secondary user in cognitive radio networks will detect available spectrum holes (i.e., unoccupied spectrum), analyze and learn information from the local environment, account for user preferences and demands, and reconfigure itself by adjusting system parameters under certain policies and regulations [6,7]. Hence, two fundamental components for cognitive radios are:

- Spectrum sensing: secondary users monitor local spectrum and detect spectrum holes when primary users are inactive; and
- Spectrum access: secondary users opportunistically utilize the spectrum holes in an intelligent manner while not interfering with primary users.



(a) Ad Hoc Cognitive Radio Networks

(b) Centralized Cognitive Radio Networks

Figure 1.1: Topology of cognitive radio networks.

As illustrated in Fig. 1.1, cognitive radio networks can be classified into two categories: ad hoc and centralized cognitive radio networks. In ad hoc cognitive radio networks, secondary users communicate with each other in a distributed fashion due to the lack of a centralized controller, e.g., a secondary base station or a fusion center. By exchanging local observed information, secondary users may gain the knowledge of network status. Usually, distributed algorithms [8] are required among the secondary users to conduct a common task, e.g., distributed spectrum sensing [9] and distributed spectrum access [10]. On the contrary, in centralized cognitive radio networks, the secondary users directly communicate to the centralized controller and report the local observed information to it. The centralized controller then makes a global decision on how to avoid interfering with primary networks [11].

In ad hoc cognitive radio networks, several recent studies have proposed and designed a common control channel [12]. This channel can be used by the individual cognitive terminals to report and share spectrum sensing results, conduct distributed spectrum sharing, etc.. Since cognitive radio may be equipped with a single transceiver, and may communicate over one of several available frequency bands, several schemes (e.g., cognitive media access control (MAC) [13]) propose using a prescribed time slot (e.g. 20ms every 100ms) on a common band as the control channel. This requires the distributed nodes be synchronized to a common notion of time.

In this dissertation, we mainly focus on two important components in cognitive radio networks: distributed time synchronization in ad hoc cognitive radio networks and cooperative spectrum sensing in centralized cognitive radio networks.

1.1.2 Time Synchronization in Cognitive Radio Networks

In a variety of applications of ad hoc cognitive radio networks, secondary users are required to maintain accurate time synchronization, e.g., solving multi-channel hidden terminal problem [14], time slotted MAC protocols [11], etc.. This necessitates network algorithms that achieve and maintain *global time synchronization* at all secondary users, i.e., algorithms that align all secondary users to a common notion of time.

Due to decentralized nature of ad hoc cognitive radio networks, global time synchronization has been recognized as a particularly challenging task. Conventional synchronization protocols such as time-synchronization protocol for sensor networks (TPSN) [15], reference broadcast synchronization (RBS) [16] and flooding time synchronization protocol (FTSP) [17] aim to perform *centralized* global synchronization for all nodes in the wireless sensor network [18]. These protocols achieve synchronization via time-stamped packet exchanges with a root node or a data fusion center and are thus vulnerable to failure of these central nodes.

Recently, several distributed time synchronization algorithms have been proposed. One important class of such algorithms is referred to as *distributed consensus time synchronization* (DCTS) [19]. Such algorithms apply the general methods of achieving consensus or agreement between distributed nodes [20] to the global timing synchronization algorithm. In the DCTS approach, a global time consensus can be sufficiently reached within a connected network by averaging pair-wise local time information at network nodes. In [21], Olfati-Saber and Murray established a theoretical framework for the analysis of consensus synchronization algorithms using continuous and discrete time models, with the presence/absence of (deterministic) time delays, and assuming both a directed and undirected network. Later, a fully distributed, asynchronous DCTS algorithm was proposed in [22]; this scheme was designed to reach agreement on time offset and skew offset between network nodes using MAC layer time-stamped packet exchanges. As an alternative, a physical (PHY) layer based DCTS algorithm was introduced in [23] by modeling sensor nodes as coupled discrete time oscillators. In particular, the algorithm adopts instantaneous received powers as weighted coefficients when updating local clocks.

1.1.3 Spectrum Sensing in Cognitive Radio Networks

To enable the dynamic spectrum access in cognitive radio networks, secondary user needs to continuously monitor local spectrum and detect spectrum holes [3, 24]. This technique, called *spectrum sensing*, requires that the secondary user reliably detect weak signals from primary users in order to avoid harmful interference. However, due to the nature of the wireless channel (e.g., fading), a secondary user may not be able to reliably differentiate between a white space and a weak primary signal if it conducts spectrum sensing on its own. To improve detection reliability, multiple secondary users can engage in cooperative spectrum sensing and thus take advantage of spatial diversity [25–27]. The topology of cooperative spectrum sensing scenario in centralized cognitive radio networks is illustrated in Fig. 1.1(b). For example, multiple secondary users can report spectrum measurements to a fusion center, where the measurements are combined to make global decisions about primary spectrum occupancy.

As described in [28, 29], cooperative spectrum sensing can be divided into two stages: local statistic/decision update stage and global fusion stage. In the first stage, secondary users process received signals to either make a local decision or compute a local statistic based on the observed information. In the global fusion stage, secondary users transmit the local statistic/decision update to a fusion center over wireless links. The fusion center combines the updates and makes a global decision. As shown in Fig. 1.2, two channel scenarios can be utilized in distributed detection: parallel access channel (PAC) [30–32] and multiple access channel (MAC) [33–35]. For PAC, the secondary users communicate to the fusion center via orthogonal channels, while for MAC, the secondary users share a common channel during transmission. The latter scheme can significantly reduce the communication bandwidth and cost, but in general requires perfect synchronization among the secondary users.

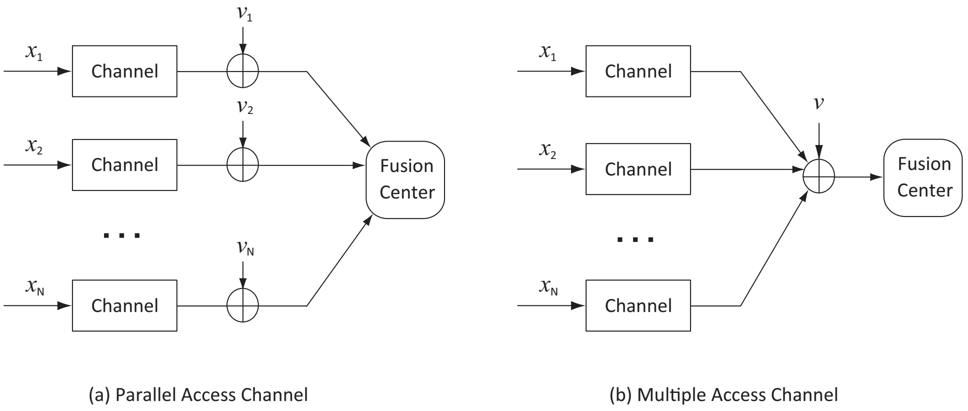


Figure 1.2: Channel scenarios for distributed detection.

In [36], it was shown that cooperative spectrum sensing under soft information combining outperforms hard decision combining. In hard decision combining, secondary users forward hard (binary) decisions based on their local spectrum measurements. In [37], a logic “OR” fusion rule for hard-decision combining was presented to cooperatively detect the primary user. The amplify-and-forward cooperative strategy was used in [38] to improve overall spectrum agility by allowing two secondary users to communicate with each other.

In soft decision combining, secondary users calculate soft information regarding the presence of a primary signal. An optimal linear detector for cooperative spectrum sensing was proposed in [25], where the received signals at the fusion center were assigned different weights for global fusion and convex optimization was formulated to find the linear weights. In [39], a linear quadratic fusion rule based on a deflection criterion was proposed for cooperative spectrum sensing by modeling received signals as correlated log-normal random variables. Low-energy overhead cooperative spectrum sensing was studied in [40], where sensing reporting overhead was reduced and optimal power allocation was computed for cooperative spectrum sensing.

1.2 Outline of the Dissertation

As previously mentioned, this dissertation mainly focuses on two important components in cognitive radio networks: distributed time synchronization in ad hoc cognitive radio networks and cooperative spectrum sensing in centralized cognitive radio networks. A brief outline of this dissertation is presented as follows.

In Chapter 2, we apply the principles of distributed consensus building to achieve global time synchronization at all secondary users. Specifically, we propose a novel discrete time *second*- and *high-order* DCTS algorithm and examine their convergence properties. Moreover, we investigate the convergence region and optimal convergence rate of the second- and high-order DCTS algorithm in undirected networks. We claim that the optimal convergence rate of the second- and high-order DCTS algorithm is superior to that of the first-order DCTS algorithm under an appropriate algorithm design.

In Chapter 3, we extend our study on the convergence of the DCTS algorithm when both deterministic and uncertain time delays impact local pair-wise time information exchange. Specifically, we model random delay between secondary users using a Gaussian approximation and determine the resulting asymptotic expectation of global synchronization error. Our results lead to the definition of a *time* *delay balanced network* and we claim that under such network topologies average timing consensus between secondary users can be achieved despite the presence of random delays. Additionally, we show that the asymptotic mean square synchronization error is lower and upper bounded by several values related to network parameters. These results, while presented here in this dissertation within the context of cognitive radio framework, can be extended to time synchronization in any distributed scenario.

In Chapter 4, we study cooperative spectrum sensing in cost constrained cognitive radio networks over PAC. In particular, we account for two major factors that contribute to the *system level* energy cost of sensing: Local processing (sample collection and local energy calculation) and transmission (forwarding local statistic to the fusion center). We devise two optimization problems to maximize the global detection probability by choosing the appropriate number of (energy) samples that must be collected at each secondary user and the appropriate amplifier gain that each secondary user must use to forward its statistics to the fusion center. When jointly designing the number of samples and amplifier gains, we demonstrate that only *one* secondary user needs to be actively processing and transmitting local statistics to the fusion center, i.e., only *one* secondary user must conduct spectrum sensing to achieve optimal performance. When either the amplifier gains or the number of samples are fixed, we derive closed-form expressions for optimal solutions and propose a generalized water-filling approach for cost constrained cooperative spectrum sensing.

In Chapter 5, we study cooperative spectrum sensing in cost constrained cognitive radio networks over MAC. In particular, we counter unreliable uplinks to the fusion center by applying beamforming amongst secondary users who communicate locally measured SNRs to a common fusion center. Under correlated lognormal shadowing, we derive optimal beamforming weights that maximize the global detection probability subject to a global transmit power constraint. We then compute the detection performance for a simplified linear array network and show that detection probability increases as the number of secondary users increases.

In this dissertation, we use the following notation: column vectors are denoted by boldface lowercase letters, i.e., $\boldsymbol{x} = [x_1, x_2, \cdots, x_{\mathrm{N}}]^{\mathsf{T}}$ and x_i is the ith entry of $\boldsymbol{x}$. $\mathbf{0} = [0, 0, \cdots, 0]^{\mathsf{T}}$ and $\mathbf{1} = [1, 1, \cdots, 1]^{\mathsf{T}}$. $(\cdot)^{\mathsf{T}}$ and $(\cdot)^{\dagger}$ denote the transpose and conjugate transpose operation, respectively. $\|\boldsymbol{x}\|$ denotes the ℓ_2 norm of $\boldsymbol{x}$. $\boldsymbol{x} \succeq \mathbf{0}$ denotes the generalized inequality, i.e., $x_i \geq 0$. $\mathcal{Z}_+^{\mathrm{N}}$ and $\mathcal{R}_+^{\mathrm{N}}$ denote the set of nonnegative integer and real n-vectors, respectively. $|\mathcal{S}|$ denotes the cardinality of a set $\mathcal{S}$. $\lceil \cdot \rceil$ and $\lfloor \cdot \rfloor$ denote the ceiling and floor operations, respectively. $[\mathbf{A}]_{ij}$ denotes the element in the ith row and jth column of $\mathbf{A}$. $\mathbf{I}_{\mathrm{N}}$ denotes the $\mathrm{N} \times \mathrm{N}$ identity matrix. $\mathbf{0}_{\mathrm{M}\times\mathrm{N}}$ denotes the $\mathrm{M} \times \mathrm{N}$ all zero matrix. $\mathbf{tr}(\cdot)$ and $\mathbf{det}(\cdot)$ denote the trace and determinant of a matrix, respectively. $\otimes$ denotes the Kronecker product. $x \sim \mathcal{N}(\mu, \sigma^2)$ means that x is a Gaussian random variable with mean μ and variance σ^2. $\mathbb{E}\{\cdot\}$ and $\mathbb{V}\mathrm{ar}\{\cdot\}$ denote the expected and variance value, respectively.

Chapter 2

Distributed Consensus Time Synchronization (DCTS) in Cognitive Radio Networks

2.1 Introduction

In this chapter, we apply the principles of distributed consensus to achieve global time synchronization at all secondary users. To the best of our knowledge, existing literature on DCTS methods assumes local timing update at each node is done using only current timing information, i.e., via a *first-order* DCTS approach. In contrast, a *second-order* DCTS algorithm would utilize not only current timing information but that available from the previous iteration of the algorithm to update local clocks. Such an extension to the basic consensus algorithm was first reported for a continuous time approach in [41]. Subsequent papers have analyzed this second-order continuous time consensus method assuming fixed network topologies [42], time delay [43] and switching topologies [44]. More generally, a high-order consensus algorithm (such as the general method reported in [45] for a continuous time approach) could be extended to update local clocks at secondary users.

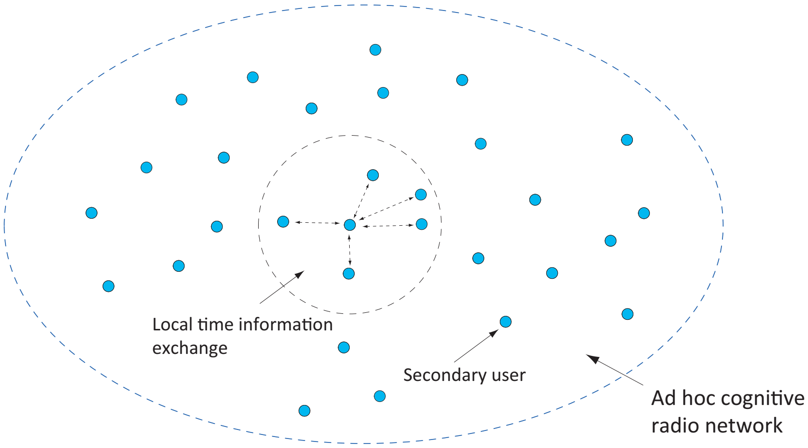

Local time information exchange
Secondary user
Ad hoc cognitive radio network

Figure 2.1: Topology of ad hoc cognitive radio networks.

Fig. 2.1 illustrates the topology of ad hoc cognitive radio networks. In this chapter, we apply the principles of the consensus approach to the distributed timing synchronization problem in ad hoc cognitive radio networks. Specifically, we propose a novel discrete time *second*- and *high-order* DCTS algorithm for ad hoc cognitive radio networks and examine their convergence properties. Moreover, we investigate the convergence region and optimal convergence rate of the second- and high-order DCTS algorithm in undirected networks. In particular, we derive closed-form expressions for those of the second-order DCTS algorithm in undirected networks. We claim that the optimal convergence rate of the second- and high-order DCTS algorithm is superior to that of the first-order DCTS algorithm under an appropriate algorithm design. These results, while presented here in this dissertation within the context of cognitive radio framework, can be extended to time synchronization in any distributed scenario.

The remainder of this chapter is outlined as follows: Section 2.2 describes network model and first-order DCTS algorithm. Section 2.3 discusses high-order DCTS algorithm and analyzes its convergence properties. Simulation results are presented in Section 2.4.

2.2 Network Model and Some Preliminaries

2.2.1 Network Model

In the following, we model an ad hoc cognitive radio network as a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, consisting of a set of N secondary users $\mathcal{V} = \{1, 2, \cdots, \mathrm{N}\}$ and a set of edges $\mathcal{E}$. Each edge is denoted as $e = (i, j) \in \mathcal{E}$ where $i \in \mathcal{V}$ and $j \in \mathcal{V}$ are head and tail of the edge e, respectively. In a cognitive radio network, the presence of an edge (i, j) indicates that secondary user i can communicate with secondary user j reliably. We assume here a connected graph; that is, there exists a directed path connecting any pair of distinct secondary users in the network. In the following, we assume a time invariant and connected network unless otherwise stated.

Given this network model, we denote $\mathbf{A}$ as the adjacency matrix of $\mathcal{G}$ such that

$$[\mathbf{A}]_{ij} = \begin{cases} 1 & (i,j) \in \mathcal{E} \\ 0 & \text{otherwise.} \end{cases} \tag{2.1}$$

Then, the in-degree and out-degree of a secondary user i (denoted as c_i and d_i, respectively) are given as $c_i = \sum_{j=1}^{\mathrm{N}}[\mathbf{A}]_{ji}$ and $d_i = \sum_{j=1}^{\mathrm{N}}[\mathbf{A}]_{ij}$. Specifically, d_i is also equal to the number of neighbors of secondary user i from which it can receive information reliably, i.e., $d_i = |\mathcal{N}_i|$, where $\mathcal{N}_i$ is the set of neighboring secondary users that can communicate reliably with secondary user i.

Next, we let $\mathbf{L}$ be the graph Laplacian matrix of $\mathcal{G}$ which is defined as $\mathbf{L} = \mathbf{D} - \mathbf{A}$, where $\mathbf{D} = \mathrm{diag}\{d_1, d_2, \cdots, d_{\mathrm{N}}\}$ is the degree matrix of $\mathcal{G}$. Given this matrix $\mathbf{L}$, we can show that $\mathbf{L1} = \mathbf{0}$. In particular, for a connected graph, the rank of $\mathbf{L}$ is $\mathrm{N} - 1$. Furthermore, for a balanced directed network, the in-degree and out-degree of a secondary user are equal, i.e., $c_i = d_i$, thus we see that $\mathbf{1}^{\mathrm{T}}\mathbf{L} = \mathbf{0}^{\mathrm{T}}$.

For an undirected network, the presence of an edge (i, j) indicates that secondary users i and j can communicate with each other reliably. Under this condition, we can also show that $\mathbf{1}^{\mathrm{T}}\mathbf{L} = \mathbf{0}^{\mathrm{T}}$. Additionally, $\mathbf{L}$ is a symmetric positive semidefinite matrix (implying its eigenvalues are non-negative); and its eigenvalues can be arranged in increasing order as $0 = \lambda_1(\mathbf{L}) < \lambda_2(\mathbf{L}) \leq \cdots \leq \lambda_{\mathrm{N}}(\mathbf{L})$ [46].

2.2.2 First-Order DCTS Algorithm

In each iteration of the first-order DCTS algorithm, each secondary user processes and decodes the time-stamped message from its neighbors in the MAC layer based approach or estimates the arrival time of its neighbors' pulse signals in the PHY layer scheme. Each secondary user then updates its local clock time using the weighted average of the time differences with its neighbor secondary users. The timing update rule of the first-order DCTS algorithm at each secondary user i is given as

$$t_i(k) \;=\; t_i(k-1) + \varepsilon \sum_{j \in \mathcal{N}_i} \big[t_j(k-1) - t_i(k-1)\big], \tag{2.2}$$

where $t_i(k)$ is the local time at secondary user i during iteration k and ε is the constant step size for each iteration[1]. Let us define $\boldsymbol{t}(k) = [t_1(k), t_2(k), \cdots, t_{\mathrm{N}}(k)]^{\mathsf{T}}$. Based on this definition, the evolution of DCTS algorithm in (2.2) can be written as

$$\boldsymbol{t}(k) \;=\; \tilde{\mathbf{H}}\boldsymbol{t}(k-1), \tag{2.3}$$

where $\tilde{\mathbf{H}} = \mathbf{I}_{\mathrm{N}} - \varepsilon\mathbf{L}$ is called a Perron matrix of a graph with parameter ε [21]. The eigenvalues of $\tilde{\mathbf{H}}$ are $\lambda_i(\tilde{\mathbf{H}}) = 1 - \varepsilon\lambda_i(\mathbf{L})$, and can be arranged in decreasing order as $1 = \lambda_1(\tilde{\mathbf{H}}) > \lambda_1(\tilde{\mathbf{H}}) \geq \cdots \geq \lambda_{\mathrm{N}}(\tilde{\mathbf{H}})$. Moreover, it is easy to show that in undirected networks, $\mathbf{1}^{\mathsf{T}}\tilde{\mathbf{H}} = \mathbf{1}^{\mathsf{T}}$ and $\tilde{\mathbf{H}}\mathbf{1} = \mathbf{1}$, hence, $\mathbf{1}$ is the left- and right-hand eigenvector of $\tilde{\mathbf{H}}$ corresponding to $\lambda_1(\tilde{\mathbf{H}}) = 1$. Additionally, we see that (2.3) can be solved by $\boldsymbol{t}(k) = \tilde{\mathbf{H}}^k\boldsymbol{t}(0)$. The known average consensus property of the first-order DCTS algorithm is then described as follows [47]:

Theorem 2.2.1. *For a time invariant, connected undirected network, when $\varepsilon \in (0, 2/\lambda_{\mathrm{N}}(\mathbf{L}))$, average consensus is asymptotically achieved with DCTS algorithm in (2.3), or equivalently,*

$$\lim_{k\to\infty} \tilde{\mathbf{H}}^k \;=\; (1/\mathrm{N})\mathbf{1}\mathbf{1}^{\mathsf{T}}. \tag{2.4}$$

The proof of this theorem is available in various sources, e.g., [47, 48]. Here we present an alternative approach to prove this theorem.

Proof. Since $\tilde{\mathbf{H}}$ is a real symmetric, doubly stochastic matrix, it is diagonalizable.

[1] We assume a constant value of ε throughout this chapter unless otherwise stated. The discussion presented here can be readily extended to the non-constant case.

Let us define the spectrum of $\tilde{\mathbf{H}}$ as $\sigma(\tilde{\mathbf{H}}) = \{\lambda_1(\tilde{\mathbf{H}}), \lambda_2(\tilde{\mathbf{H}}), \cdots, \lambda_l(\tilde{\mathbf{H}})\}$, which is the set of distinct eigenvalues. According to [49], $\tilde{\mathbf{H}}^k$ can be decomposed as

$$\tilde{\mathbf{H}}^k \;=\; \lambda_1^k(\tilde{\mathbf{H}})\mathbf{G}_1 + \lambda_2^k(\tilde{\mathbf{H}})\mathbf{G}_2 + \cdots + \lambda_l^k(\tilde{\mathbf{H}})\mathbf{G}_l,$$

where

$$\mathbf{G}_i \;=\; \prod_{j=1,j\neq i}^{l} \left[\tilde{\mathbf{H}} - \lambda_i(\tilde{\mathbf{H}})\mathbf{I}_{\mathrm{N}}\right] / \prod_{j=1,j\neq i}^{l} \left[\lambda_i(\tilde{\mathbf{H}}) - \lambda_j(\tilde{\mathbf{H}})\right].$$

If $\lambda_i(\tilde{\mathbf{H}})$ is an eigenvalue associated with right- and left-hand eigenvectors $\boldsymbol{x}$ and $\boldsymbol{y}$, respectively, then $\mathbf{G}_i = \boldsymbol{x}\boldsymbol{y}^{\mathrm{T}}/\boldsymbol{y}^{\mathrm{T}}\boldsymbol{x}$. It is easy to show that, $\mathbf{G}_1 = (1/\mathrm{N})\mathbf{1}\mathbf{1}^{\mathrm{T}}$ is associated with eigenvalue $\lambda_1(\tilde{\mathbf{H}}) = 1$.

When $0 < \varepsilon < 2/\lambda_{\mathrm{N}}(\mathbf{L})$, we have $|1 - \varepsilon\lambda_i(\mathbf{L})| < 1,\ i = 2, \cdots, \mathrm{N}$. Thus, $\lim_{k\to\infty} \lambda_i^k(\tilde{\mathbf{H}}) = 0,\ i = 2, \cdots, \mathrm{N}$. Then, for a stable system, we have

$$\lim_{k\to\infty} \tilde{\mathbf{H}}^k \;=\; \lim_{k\to\infty} \lambda_1^k(\tilde{\mathbf{H}})\mathbf{G}_1 \;=\; \mathbf{G}_1 \;=\; (1/\mathrm{N})\mathbf{1}\mathbf{1}^{\mathrm{T}}.$$

This completes the proof. □

Let us define the convergence rate of the DCTS algorithm as

$$\nu \;=\; \sup_{\boldsymbol{t}(0)\neq\bar{t}\mathbf{1}} \lim_{k\to\infty} \left(\frac{\|\boldsymbol{t}(k) - \bar{t}\mathbf{1}\|}{\|\boldsymbol{t}(0) - \bar{t}\mathbf{1}\|}\right)^{1/k},$$

where $\bar{t} = (1/\mathrm{N})\mathbf{1}^{\mathrm{T}}\boldsymbol{t}(0)$. As shown in [48], the convergence rate of the first-order DCTS algorithm can be computed as $\nu = \rho\big(\tilde{\mathbf{H}} - (1/\mathrm{N})\mathbf{1}\mathbf{1}^{\mathrm{T}}\big)$, where $\rho(\cdot)$ denotes the spectral radius of a matrix. Furthermore, when ε is chosen as

$$\varepsilon_{\text{opt, FO}} = \frac{2}{\lambda_{\text{N}}(\mathbf{L}) + \lambda_2(\mathbf{L})}, \tag{2.5}$$

the optimal convergence rate of the first-order DCTS algorithm can be achieved as

$$\nu_{\text{opt, FO}} = \frac{\lambda_{\text{N}}(\mathbf{L}) - \lambda_2(\mathbf{L})}{\lambda_{\text{N}}(\mathbf{L}) + \lambda_2(\mathbf{L})}.$$

2.3 High-Order DCTS Algorithm

In this section, we describe the high-order DCTS method regardless of whether it is implemented at the PHY or MAC layers. In each iteration of the high-order DCTS algorithm, each secondary user processes and decodes the time-stamped message from its neighbors or estimates the arrival time of its neighbors' pulse signals. Each secondary user then updates its local clock using the weighted average of the current time differences between itself and its neighboring secondary users as well as stored time differences from the $\text{M} - 1$ previous iterations of the algorithm. It should be noted that in the M-*th* order DCTS algorithm, each secondary user needs to store time information from all its neighbors for the past $\text{M} - 1$ iterations as well as the current iteration; this is in contrast to the first-order DCTS approach where only the current time information is processed in the current iteration.

The timing update rule of the M-*th* order DCTS algorithm at each secondary user i is proposed as

$$\begin{aligned} t_i(k) &= t_i(k-1) + \varepsilon \sum_{m=0}^{\mathrm{M}-1} c_m(-\gamma)^m \Delta t_i(k,m) \\ \Delta t_i(k,m) &= \sum_{j\in\mathcal{N}_i} \big[t_j(k-m-1) - t_i(k-m-1)\big], \end{aligned} \tag{2.6}$$

where c_m are predefined constants with $c_0 = 1$ and $c_m \neq 0$ $(m > 0)$; and γ is a forgetting factor and $|\gamma| < 1$. We assume initial conditions of the M-*th* order DCTS algorithm are $t_i(-\mathrm{M}+1) = \cdots = t_i(-1) = t_i(0) = \xi_i$, where ξ_i is initial time offset for secondary user i.

It is worth mentioning that when $\gamma = 0$, the high-order DCTS algorithm reduces to the first-order DCTS algorithm; while when $\mathrm{M} = 2$, the high-order DCTS algorithm reduces to second-order DCTS algorithm, i.e.,

$$\begin{aligned} t_i(k) &= t_i(k-1) + \varepsilon \sum_{j\in\mathcal{N}_i} \big[t_j(k-1) - t_i(k-1)\big] \\ &\quad - \varepsilon c_1 \gamma \sum_{j\in\mathcal{N}_i} \big[t_j(k-2) - t_i(k-2)\big]. \end{aligned} \tag{2.7}$$

The evolution of the high-order DCTS algorithm in (2.6) can be written as

$$\boldsymbol{t}(k) = (\mathbf{I}_\mathrm{N} - \varepsilon\mathbf{L})\boldsymbol{t}(k-1) - \varepsilon \sum_{m=1}^{\mathrm{M}-1} c_m(-\gamma)^m \mathbf{L}\boldsymbol{t}(k-m-1), \tag{2.8}$$

with the initial conditions $\boldsymbol{t}(-\mathrm{M}+1) = \cdots = \boldsymbol{t}(-1) = \boldsymbol{t}(0) = \boldsymbol{\xi}$, where $\boldsymbol{\xi} = [\xi_1, \xi_2, \cdots, \xi_\mathrm{N}]^\mathtt{T}$.

2.3.1 Convergence Analysis of High-Order DCTS Algorithm

(1) Convergence Properties of High-Order DCTS Algorithm

Before we investigate the convergence properties of the high-order DCTS algorithm, we define two MN × MN matrices

$$\mathbf{H} = \begin{bmatrix} \mathbf{I}_{\mathrm{N}} - \varepsilon\mathbf{L} & c_1\gamma\varepsilon\mathbf{L} & \cdots & -c_{\mathrm{M}-1}(-\gamma)^{\mathrm{M}-1}\varepsilon\mathbf{L} \\ \mathbf{I}_{\mathrm{N}} & \mathbf{0}_{\mathrm{N}\times\mathrm{N}} & \cdots & \mathbf{0}_{\mathrm{N}\times\mathrm{N}} \\ \vdots & \ddots & & \vdots \\ \mathbf{0}_{\mathrm{N}\times\mathrm{N}} & \cdots & \mathbf{I}_{\mathrm{N}} & \mathbf{0}_{\mathrm{N}\times\mathrm{N}} \end{bmatrix}$$

and

$$\mathbf{J} = \begin{bmatrix} \mathbf{K} & \mathbf{0}_{\mathrm{N}\times\mathrm{N}} & \cdots & \mathbf{0}_{\mathrm{N}\times\mathrm{N}} \\ \mathbf{K} & \mathbf{0}_{\mathrm{N}\times\mathrm{N}} & \cdots & \mathbf{0}_{\mathrm{N}\times\mathrm{N}} \\ \vdots & \ddots & & \vdots \\ \mathbf{K} & \mathbf{0}_{\mathrm{N}\times\mathrm{N}} & \cdots & \mathbf{0}_{\mathrm{N}\times\mathrm{N}} \end{bmatrix},$$

where $\mathbf{K} = \mathbf{1}\boldsymbol{\beta}^{\mathrm{T}}/(\boldsymbol{\beta}^{\mathrm{T}}\mathbf{1})$ and $\boldsymbol{\beta}$ is the left eigenvector of $\mathbf{L}$ associated with $\lambda_1(\mathbf{L}) = 0$, i.e., $\boldsymbol{\beta}^{\mathrm{T}}\mathbf{L} = \mathbf{0}^{\mathrm{T}}$. Then we have the following lemma:

Lemma 2.3.1. *The eigenvalues of* $\mathbf{H}-\mathbf{J}$ *agree with those of* $\mathbf{H}$ *except that* $\lambda_1(\mathbf{H}) = 1$ *is replaced by* $\lambda_1(\mathbf{H}-\mathbf{J}) = 0$.

Proof. Let us define two MN × 1 vectors $\boldsymbol{h}_l = (1/\boldsymbol{\beta}^{\mathrm{T}}\mathbf{1})[\boldsymbol{\beta}^{\mathrm{T}}\ \mathbf{0}^{\mathrm{T}}\ \cdots\ \mathbf{0}^{\mathrm{T}}]^{\mathrm{T}}$ and $\boldsymbol{h}_r = [\mathbf{1}^{\mathrm{T}}\ \cdots\ \mathbf{1}^{\mathrm{T}}\ \mathbf{1}^{\mathrm{T}}]^{\mathrm{T}}$. It is easy to check that $\boldsymbol{h}_l$ and $\boldsymbol{h}_r$ are left and right eigenvectors of $\mathbf{H}$ corresponding to $\lambda_1(\mathbf{H}) = 1$, respectively, i.e., $\boldsymbol{h}_l^{\mathrm{T}}\mathbf{H} = \boldsymbol{h}_l^{\mathrm{T}}$ and $\mathbf{H}\boldsymbol{h}_r = \boldsymbol{h}_r$.

Additionally, $\mathbf{J} = \boldsymbol{h}_r \boldsymbol{h}_l^{\mathsf{T}}$, $\boldsymbol{h}_l^{\mathsf{T}} \boldsymbol{h}_r = \boldsymbol{h}_r^{\mathsf{T}} \boldsymbol{h}_l = 1$. In order to obtain the eigenvalues of $\mathbf{H} - \mathbf{J}$, we have [49]

$$\begin{aligned} \mathbf{det}\left(\mathbf{H} - \mathbf{J} - \lambda \mathbf{I}_{\mathrm{MN}}\right) &= \mathbf{det}\left(\mathbf{H} - \boldsymbol{h}_r \boldsymbol{h}_l^{\mathsf{T}} - \lambda \mathbf{I}_{\mathrm{MN}}\right) \\ &= \mathbf{det}\left(\mathbf{H} - \lambda \mathbf{I}_{\mathrm{MN}}\right)\left[1 - \boldsymbol{h}_l^{\mathsf{T}} \left(\mathbf{H} - \lambda \mathbf{I}_{\mathrm{MN}}\right)^{-1} \boldsymbol{h}_r\right] \\ &= \left[\pm \prod_{i=1}^{\mathrm{MN}} \left(\lambda_i(\mathbf{H}) - \lambda\right)\right]\left(1 - \frac{\boldsymbol{h}_l^{\mathsf{T}} \boldsymbol{h}_r}{1 - \lambda}\right) \\ &= \left[\pm \prod_{i=2}^{\mathrm{MN}} \left(\lambda_i(\mathbf{H}) - \lambda\right)\right](-\lambda). \end{aligned} \tag{2.9}$$

The above equation is valid because

$$\boldsymbol{h}_r = (\mathbf{H} - \lambda \mathbf{I}_{\mathrm{MN}})^{-1} (\mathbf{H} - \lambda \mathbf{I}_{\mathrm{MN}}) \boldsymbol{h}_r = (\mathbf{H} - \lambda \mathbf{I}_{\mathrm{MN}})^{-1} (1 - \lambda) \boldsymbol{h}_r.$$

Therefore, the eigenvalues of $\mathbf{H} - \mathbf{J}$ are:

$$\lambda_1(\mathbf{H} - \mathbf{J}) = 0 \quad \text{and} \quad \lambda_i(\mathbf{H} - \mathbf{J}) = \lambda_i(\mathbf{H}), \quad i = 2, \cdots, \mathrm{MN}.$$

This completes the proof. □

Our main result regarding the global consensus property of the M-*th* order DCTS algorithm in directed networks is stated in the following theorem.

Theorem 2.3.1. *For a time invariant, connected, directed network, consider the* M-th *order DCTS algorithm,*

$$\boldsymbol{t}(k) = (\mathbf{I}_{\mathrm{N}} - \varepsilon \mathbf{L}) \boldsymbol{t}(k-1) - \varepsilon \sum_{m=1}^{\mathrm{M}-1} c_m (-\gamma)^m \mathbf{L} \boldsymbol{t}(k-m-1), \tag{2.10}$$

with initial conditions $\boldsymbol{t}(-\mathrm{M}+1) = \cdots = \boldsymbol{t}(-1) = \boldsymbol{t}(0) = \boldsymbol{\xi}$. *When* $\rho(\mathbf{H} - \mathbf{J}) < 1$, *a global consensus is achieved asymptotically, or equivalently,*

$$\lim_{k\to\infty} t_i(k) \;=\; \frac{\boldsymbol{\beta}^T\boldsymbol{\xi}}{\boldsymbol{\beta}^T\mathbf{1}}, \quad \forall i \in \mathcal{V}.$$

Proof. The proof of this theorem is similar to [45, 50]. Here we present a sketch proof. Let us define $\boldsymbol{\psi}(k) = \left[\boldsymbol{t}(k)^{\mathrm{T}}\ \boldsymbol{t}(k-1)^{\mathrm{T}}\ \cdots\ \boldsymbol{t}(k-\mathrm{M}+1)^{\mathrm{T}}\right]^{\mathrm{T}}$. Then, the M-*th* order DCTS algorithm in (2.10) can be rewritten as

$$\boldsymbol{\psi}(k) \;=\; \mathbf{H}\boldsymbol{\psi}(k-1), \tag{2.11}$$

which implies $\boldsymbol{\psi}(k) = \mathbf{H}^k\boldsymbol{\psi}(0)$. To calculate the eigenvalues of $\mathbf{H}$, we have [49]:

$$\begin{aligned}
&\mathbf{det}\left(\mathbf{H} - \lambda\mathbf{I}_{\mathrm{MN}}\right)\\
=\;& \mathbf{det}\left(\lambda^{\mathrm{M}}\mathbf{I}_{\mathrm{N}} - (\mathbf{I}_{\mathrm{N}} - \varepsilon\mathbf{L})\lambda^{\mathrm{M}-1} + \varepsilon\sum_{m=1}^{\mathrm{M}-1} c_m(-\gamma)^m\lambda^{\mathrm{M}-1-m}\mathbf{L}\right)\\
=\;& \prod_{i=1}^{\mathrm{N}}\left[\lambda^{\mathrm{M}} - (1 - \varepsilon\lambda_i(\mathbf{L}))\lambda^{\mathrm{M}-1} + \varepsilon\sum_{m=1}^{\mathrm{M}-1} c_m(-\gamma)^m\lambda_i(\mathbf{L})\lambda^{\mathrm{M}-1-m}\right]\\
=\;& 0.
\end{aligned}$$

Thus, the eigenvalues of $\mathbf{H}$ should satisfy the following equation:

$$f(\lambda) \;=\; \lambda^{\mathrm{M}} - (1 - \varepsilon\lambda_i(\mathbf{L}))\lambda^{\mathrm{M}-1} + \varepsilon\sum_{m=1}^{\mathrm{M}-1} c_m(-\gamma)^m\lambda_i(\mathbf{L})\lambda^{\mathrm{M}-1-m} \;=\; 0. \tag{2.12}$$

Note that there are M roots corresponding to one $\lambda_i(\mathbf{L})$, which implies M eigenvalues of $\mathbf{H}$ are associated with one eigenvalue of $\mathbf{L}$. For a time invariant, connected, directed network, $\mathbf{L}$ has only one eigenvalue $\lambda_1(\mathbf{L}) = 0$. From (2.12), when $\lambda_1(\mathbf{L}) = 0$, the eigenvalues of $\mathbf{H}$ satisfy $f(\lambda) = \lambda^{\mathrm{M}} - \lambda^{\mathrm{M}-1} = 0$. Then, for this $\lambda_1(\mathbf{L}) = 0$, $\mathbf{H}$ has only two distinct eigenvalues $\lambda_1(\mathbf{H}) = 1$ (with algebraic multiplicity 1) and $\lambda_2(\mathbf{H}) = 0$ (with algebraic multiplicity $\mathrm{M} - 1$). Additionally, it is easy to show that the algebraic multiplicity of eigenvalue $\lambda(\mathbf{H}) = 1$ is equal to 1. Based on Lemma 2.3.1, we know that the eigenvalues of $\mathbf{H} - \mathbf{J}$ agree with those of $\mathbf{H}$ except that $\lambda_1(\mathbf{H}) = 1$ is replaced by $\lambda_1(\mathbf{H} - \mathbf{J}) = 0$. Since $\rho(\mathbf{H} - \mathbf{J}) < 1$, we see that the eigenvalues of $\mathbf{H}$ stay inside the unit circle except for $\lambda_1(\mathbf{H}) = 1$. Thus, we have

$$\begin{aligned}
\lim_{k\to\infty} \mathbf{H}^k &= \mathbf{V} \lim_{k\to\infty} \begin{bmatrix} 1 & \mathbf{0}_{1\times(\mathrm{MN}-1)} \\ \mathbf{0}_{(\mathrm{MN}-1)\times 1} & \mathbf{\Lambda}^k \end{bmatrix} \mathbf{V}^{-1} \\
&= \mathbf{V} \begin{bmatrix} 1 & \mathbf{0}_{1\times(\mathrm{MN}-1)} \\ \mathbf{0}_{(\mathrm{MN}-1)\times 1} & \mathbf{0}_{(\mathrm{MN}-1)\times(\mathrm{MN}-1)} \end{bmatrix} \mathbf{V}^{-1} \\
&= \boldsymbol{h}_r \boldsymbol{h}_l^{\mathrm{T}}, \qquad (2.13)
\end{aligned}$$

where $\mathbf{\Lambda}$ is the Jordan form matrix corresponding to eigenvalues $\lambda_i(\mathbf{H}) \neq 1$ [49], and $\boldsymbol{h}_l$ and $\boldsymbol{h}_r$ are defined in the proof of Lemma 2.3.1. Plugging $\boldsymbol{h}_l$ and $\boldsymbol{h}_r$ into (2.13) and considering the M-*th* order DCTS algorithm in (2.10), we have $\lim_{k\to\infty} \mathbf{H}^k = \mathbf{J}$. Then, $\lim_{k\to\infty} \boldsymbol{\psi}(k) = \mathbf{J}\boldsymbol{\psi}(0)$, which indicates

$$\lim_{k\to\infty} t_i(k) = \frac{\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\xi}}{\boldsymbol{\beta}^{\mathrm{T}}\mathbf{1}}.$$

This completes the proof. □

According to Theorem 2.3.1, we see that in general, although average consensus is not achieved for directed networks using the high-order DCTS method, all secondary users in the network can still reach a global agreement. By "average consensus" we mean that all secondary users converge to the same timing which is determined by the average of the initial timing differences between the secondary users. However, when the M-*th* order DCTS algorithm is employed in either an undirected network or a *balanced* directed network, average consensus can be achieved asymptotically. We show this via the following theorem:

Theorem 2.3.2. *Consider the* M-*th order DCTS algorithm in (2.10) in a time invariant, connected, directed balanced network or a time invariant, connected, undirected network, with initial conditions* $\boldsymbol{t}(-\mathrm{M}+1) = \cdots = \boldsymbol{t}(-1) = \boldsymbol{t}(0) = \boldsymbol{\xi}$. *When* $\rho(\mathbf{H} - \mathbf{J}) < 1$, *an average consensus is achieved asymptotically, or equivalently,*

$$\lim_{k\to\infty} t_i(k) \;=\; (1/\mathrm{N})\mathbf{1}^{\mathrm{T}}\boldsymbol{\xi}, \quad \forall i \in \mathcal{V}.$$

We know that in a time invariant, connected, directed balanced or undirected network, $\boldsymbol{\beta} = \mathbf{1}$ and $\mathbf{K} = (1/\mathrm{N})\mathbf{1}\mathbf{1}^{\mathrm{T}}$. The rest of proof is similar to that of Theorem 2.3.1 and thus omitted here.

(2) Convergence Rate for High-Order DCTS Algorithm

One of the most important measures of any distributed, iterative algorithm is its convergence speed. As we show next, the convergence rate of the high-order DCTS algorithm is determined by the spectral radius of $\mathbf{H} - \mathbf{J}$, which is similar to the first-order DCTS algorithm [47].

Let us define the global consensus value in each iteration as $m(k) = \frac{\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{t}(k)}{\boldsymbol{\beta}^{\mathrm{T}}\mathbf{1}}$. In the high-order DCTS algorithm, this value remains invariant during each iteration since

$$\begin{aligned} m(k) &= (1/\boldsymbol{\beta}^{\mathrm{T}}\mathbf{1})\boldsymbol{\beta}^{\mathrm{T}}\left[(\mathbf{I}_{\mathrm{N}} - \varepsilon\mathbf{L})\boldsymbol{t}(k-1) - \varepsilon\sum_{m=1}^{\mathrm{M}-1} c_m(-\gamma)^m\mathbf{L}\boldsymbol{t}(k-m-1)\right] \\ &= m(k-1) \;=\; \cdots \;=\; m(0). \end{aligned}$$

We now define the disagreement vector as $\boldsymbol{\delta}(k) = \boldsymbol{t}(k) - m(k)\mathbf{1}$, which indicates the difference between the updated times and the global consensus times of the secondary users. Then, the evolution of the disagreement vector is obtained as:

$$\boldsymbol{\delta}(k) \;=\; (\mathbf{I}_{\mathrm{N}} - \varepsilon\mathbf{L})\boldsymbol{\delta}(k-1) - \varepsilon\sum_{m=1}^{\mathrm{M}-1} c_m(-\gamma)^m\mathbf{L}\boldsymbol{\delta}(k-m-1). \tag{2.14}$$

Given this dynamic of the disagreement vector, we note that

Lemma 2.3.2. *For the* M*-th order DCTS algorithm in (2.10) in a time invariant, connected network with initial conditions* $\boldsymbol{t}(-\mathrm{M}+1) = \cdots = \boldsymbol{t}(-1) = \boldsymbol{t}(0) = \boldsymbol{\xi}$ *and* $\rho(\mathbf{H} - \mathbf{J}) < 1$, *a global consensus is exponentially reached in the following form:*

$$\frac{\sum_{m=0}^{\mathrm{M}-1} \|\boldsymbol{\delta}(k-m)\|^2}{\|\boldsymbol{\delta}(0)\|^2} \;\leq\; \mathrm{M}\nu^{2k}, \tag{2.15}$$

where $\nu = \rho(\mathbf{H} - \mathbf{J})$.

Proof. Let us define the error vector as $\boldsymbol{e}(k) = \left[\boldsymbol{\delta}^{\mathrm{T}}(k)\ \boldsymbol{\delta}^{\mathrm{T}}(k-1)\ \cdots\ \boldsymbol{\delta}^{\mathrm{T}}(k-\mathrm{M}+1)\right]^{\mathrm{T}}$ which can be obtained from $\boldsymbol{e}(k) = \boldsymbol{\psi}(k) - \mathbf{J}_1\boldsymbol{\psi}(k)$, where $\mathbf{J}_1 = \mathbf{I}_{\mathrm{M}} \otimes \mathbf{K}$.

Based on this definition, we see that the error vector results in the following evolution:

$$\begin{aligned}\boldsymbol{e}(k) &= (\mathbf{H} - \mathbf{J}_1\mathbf{H})\,\boldsymbol{\psi}(k-1) \\ &= (\mathbf{H} - \mathbf{J})\,[\boldsymbol{\psi}(k-1) - \mathbf{J}_1\boldsymbol{\psi}(k-1)] \\ &= (\mathbf{H} - \mathbf{J})\,\boldsymbol{e}(k-1). \end{aligned} \tag{2.16}$$

The above equation is valid because $(\mathbf{H} - \mathbf{J})\mathbf{J}_1 = \mathbf{0}_{\mathrm{MN}\times\mathrm{MN}}$ and $\mathbf{J}_1\mathbf{H} = \mathbf{J}$. Then, we have

$$\|\boldsymbol{e}(k)\|^2 \;=\; \|(\mathbf{H} - \mathbf{J})\boldsymbol{e}(k-1)\|^2 \;\leq\; \nu^2\|\boldsymbol{e}(k-1)\|^2 \;\leq\; \cdots \;\leq\; \nu^{2k}\|\boldsymbol{e}(0)\|^2,$$

which is equivalent to (2.15). This completes the proof. □

Therefore, we see that the convergence rate for the M-*th* order DCTS algorithm in both directed and undirected networks is determined by the spectral radius of $\mathbf{H} - \mathbf{J}$.

2.3.2 Convergence Region and Optimal Convergence Rate of High-Order DCTS Algorithm

In this section, we investigate more specific convergence results (i.e., the convergence region and optimal convergence rate) for the high-order DCTS algorithm in undirected networks. Without loss of generality, we assume that ε and γ are real values, and $\varepsilon > 0$.

(1) Convergence Region for High-Order DCTS Algorithm in Undirected Networks

From Theorem 2.3.2, we know that when $\rho(\mathbf{H}-\mathbf{J}) < 1$, the M-*th* order DCTS algorithm in an undirected network can achieve average consensus asymptotically. Let us define the convergence region $\mathcal{R}$ to satisfy $\rho(\mathbf{H}-\mathbf{J}) < 1$, i.e.,

$$\mathcal{R} \;=\; \{(\varepsilon,\gamma)|\rho(\mathbf{H}-\mathbf{J}) < 1\}\,. \tag{2.17}$$

It should be noted that in general, closed-form solution of the convergence region of the high-order DCTS algorithm is hard to find due to the fact that high-order polynomial equations are involved in calculating the eigenvalues of $\mathbf{H}-\mathbf{J}$. For example, when M $= 3$ and $c_1 = 1, c_2 = 1$, we need to find the roots of the following cubic equation to obtain the eigenvalues of $\mathbf{H}-\mathbf{J}$:

$$f(\lambda) \;=\; \lambda^3 - (1-\varepsilon\lambda_i(\mathbf{L}))\lambda^2 - \gamma\varepsilon\lambda_i(\mathbf{L})\lambda + \gamma^2\varepsilon\lambda_i(\mathbf{L}) \;=\; 0. \tag{2.18}$$

However, when M $= 2$, closed-form expressions of the convergence region for second-order DCTS algorithm can be obtained. Without loss of generality, we assume $c_1 = 1$ for the second-order DCTS algorithm throughout the dissertation. Using the steps outlined in the Appendix A.1, the convergence region for the second-order DCTS algorithm in undirected networks is

$$\mathcal{R} \;=\; \mathcal{R}' \cup \mathcal{R}'', \tag{2.19}$$

where

$$\mathcal{R}' = \left\{-\frac{1}{\varepsilon\lambda_{\mathrm{N}}(\mathbf{L})} < \gamma < 1,\ 0 < \varepsilon < \frac{1}{\lambda_{\mathrm{N}}(\mathbf{L})}\right\},$$

and

$$\mathcal{R}'' = \left\{-\frac{1}{\varepsilon\lambda_{\mathrm{N}}(\mathbf{L})} < \gamma < \frac{2}{\varepsilon\lambda_{\mathrm{N}}(\mathbf{L})} - 1,\ \frac{1}{\lambda_{\mathrm{N}}(\mathbf{L})} \leq \varepsilon < \frac{3}{\lambda_{\mathrm{N}}(\mathbf{L})}\right\}.$$

The convergence region of the second-order DCTS algorithm in undirected networks is shown in Fig. 2.2 and Fig. 2.3 using a three-dimensional and two-dimensional perspective, respectively. We see that compared to the first-order DCTS algorithm where the range of the step size ε is $(0, 2/\lambda_{\mathrm{N}}(\mathbf{L}))$, the range of ε in the second-order DCTS approach increases to $(0, 3/\lambda_{\mathrm{N}}(\mathbf{L}))$.

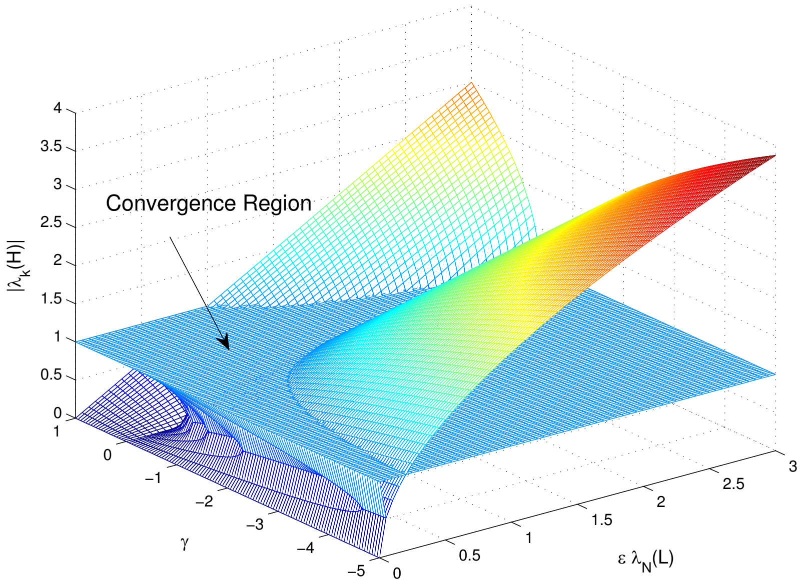


Figure 2.2: Convergence region for the second-order DCTS algorithm in undirected networks: three-dimensional view.

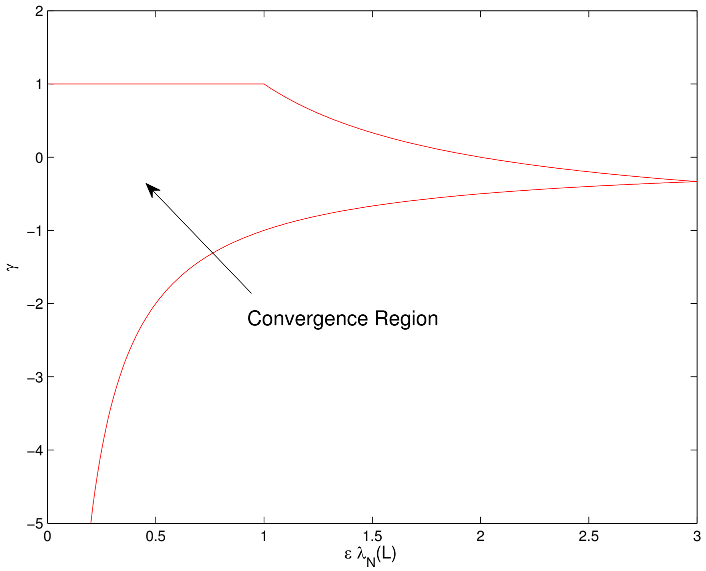


Figure 2.3: Convergence region for the second-order DCTS algorithm in undirected networks: two-dimensional view.

(2) Optimal Convergence Rate for High-Order DCTS Algorithm in Undirected Networks

Next, we investigate the fastest convergence rate of the high-order DCTS algorithm based on ε and γ. To see this, we formulate the following spectral radius minimization problem to find the optimal ε and γ for the high-order DCTS algorithm, i.e.,

$$\begin{aligned} \min_{\varepsilon,\gamma} \quad & \rho(\mathbf{H}-\mathbf{J}) \\ \text{s.t.} \quad & (\varepsilon,\gamma)\in\mathcal{R}. \end{aligned} \tag{2.20}$$

Let us define the minimal spectral radius of $\mathbf{H} - \mathbf{J}$ as

$$\nu_{\text{opt}} = \min\{\rho(\mathbf{H} - \mathbf{J})\}.$$

Again, we see that the closed form solution for this optimization problem is difficult to find when $\mathrm{M} > 2$. In practical applications, however, since the optimal ε and γ depend only on the network topology, a numerical solution can be obtained offline based on secondary user deployment and all design parameters can be flooded to the secondary users before they run the distributed algorithm.

Similar to the convergence region, when $\mathrm{M} = 2$, we can obtain closed-form solution for the optimal convergence rate of the second-order DCTS algorithm. As we show next, the convergence rate of the second-order DCTS algorithm can be superior to that of the first-order DCTS algorithm by choosing suitable ε and γ. However, as stated in the following lemma, the convergence rate of the first-order DCTS algorithm is faster under some circumstances.

Lemma 2.3.3. *For the second-order DCTS algorithm in (2.7) in a time invariant, connected, undirected network with initial conditions* $\boldsymbol{t}(-1) = \boldsymbol{t}(0) = \boldsymbol{\xi}$ *and* $(\varepsilon, \gamma) \in \mathcal{R}$ *in (2.19), if* $\gamma > 0$*, the convergence rate of the second-order DCTS algorithm is less than that of the first-order DCTS algorithm with the optimal constant step size in (2.5).*

The proof of this lemma is omitted here since it can be readily extended from the following result: Consider two real values a and b with $b > 0$, then $\max\left\{\frac{1}{2}\left|a + \sqrt{a^2 + b}\right|, \frac{1}{2}\left|a - \sqrt{a^2 + b}\right|\right\} > a$. Thus, we have $|\lambda_k(\mathbf{H})| > 1 - \varepsilon\lambda_i(\mathbf{L})$, which implies $|\lambda_k(\mathbf{H})| > \nu_{\text{opt,FO}}$.

Based on the above lemma, we see that there may exist possible choices of ε and γ (e.g., when $\gamma < 0$) such that the convergence rate of the second-order DCTS method is faster than the first-order DCTS algorithm. Using the steps outlined in the Appendix A.2 and restricting $\gamma < 0$, the optimal ε and γ for the optimization problem (2.20) can be obtained as

$$\begin{aligned}
\varepsilon_{\text{opt,SO}} &= \frac{3\lambda_{\text{N}}(\mathbf{L}) + \lambda_2(\mathbf{L})}{\lambda_{\text{N}}(\mathbf{L})\left[\lambda_{\text{N}}(\mathbf{L}) + 3\lambda_2(\mathbf{L})\right]} \\
\gamma_{\text{opt,SO}} &= -\frac{\left[\lambda_{\text{N}}(\mathbf{L}) - \lambda_2(\mathbf{L})\right]^2}{\left[\lambda_{\text{N}}(\mathbf{L}) + 3\lambda_2(\mathbf{L})\right]\left[3\lambda_{\text{N}}(\mathbf{L}) + \lambda_2(\mathbf{L})\right]}.
\end{aligned} \tag{2.21}$$

It is worth noting that $(\varepsilon_{\text{opt,SO}}, \gamma_{\text{opt,SO}}) \in \mathcal{R}''$. Then, the optimal convergence rate for the second-order DCTS algorithm in undirected networks can be calculated as

$$\nu_{\text{opt,SO}} = \frac{\lambda_{\text{N}}(\mathbf{L}) - \lambda_2(\mathbf{L})}{\lambda_{\text{N}}(\mathbf{L}) + 3\lambda_2(\mathbf{L})}. \tag{2.22}$$

We see that $\nu_{\text{opt,SO}} \leq \nu_{\text{opt,FO}}$ and $\nu_{\text{opt,SO}} = \nu_{\text{opt,FO}}$ only when $\lambda_2(\mathbf{L}) = \lambda_{\text{N}}(\mathbf{L})$. Thus, we have the following theorem for the convergence rate of the second-order DCTS algorithm.

Theorem 2.3.3. *For the second-order DCTS algorithm in (2.7) in a time invariant, connected, undirected network with initial conditions* $\boldsymbol{t}(-1) = \boldsymbol{t}(0) = \boldsymbol{\xi}$ *and* $(\varepsilon, \gamma) \in \mathcal{R}$ *in (2.19), there exists a pair of* ε *and* γ *such that the convergence rate of the second-order DCTS algorithm is greater than or equal to that of the first-order DCTS algorithm with the optimal constant step size in (2.5).*

2.4 Simulation Results

In the following, we simulate networks in which the initial time phase[2] of secondary user i is $(i-1/2)T/\mathrm{N}$, $i = 1, \cdots, \mathrm{N}$, where $T = 1000\mu s$. Specifically, we study the following two undirected network topologies:

- Case I: Fixed network with 6 secondary users as shown in Fig. 2.4(a).
- Case II: Random network with 16 secondary users. The 16 secondary users were randomly generated with uniform distribution over a unit square kilometer; two secondary users were assumed connected if the distance between them was less than η, a predefined threshold. One realization of such a network is shown in Fig. 2.4(b).

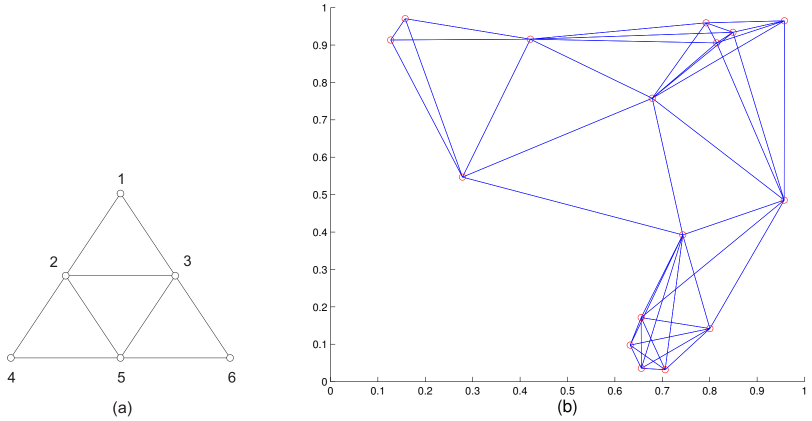


Figure 2.4: (a) Fixed network with 6 secondary users, (b) random network with 16 secondary users with $\eta = 0.5$.

[2] Trends similar to the ones noted below were observed when initial time offsets between secondary users were arbitrary (e.g., when they were uniformly distributed over $[0, T]$). We use this fixed offset assumption here for comparison purposes.

(1) Convergence Rate Comparison

For the sake of simplicity, we only consider M $= 2, 3, 4$ for the high-order DCTS approach. In the simulations, we denote our proposed DCTS algorithm as best constant (BC) high-order DCTS algorithm and choose two types of ad hoc weights as comparison: maximum degree (MD) and metropolis hasting (MH) weights [51]. Furthermore, we assume $c_1 = 1, c_2 = 1$ and $c_3 = 1/6$.

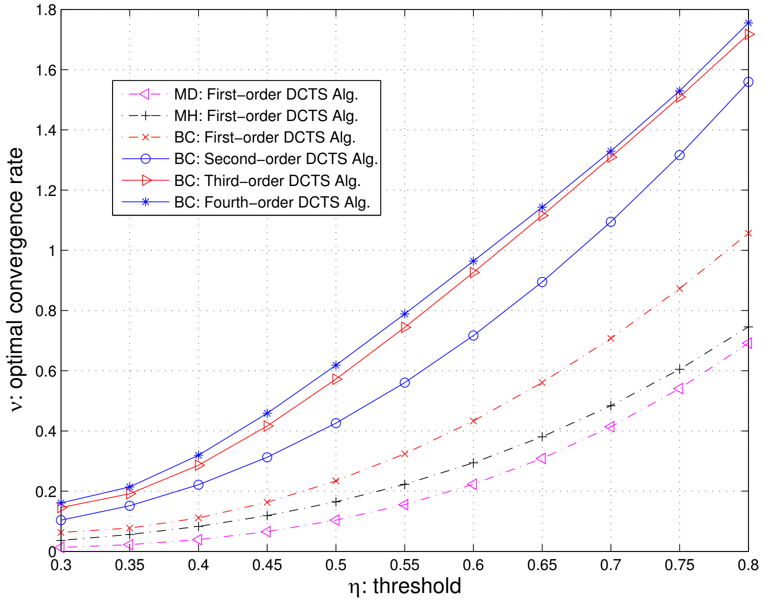


Figure 2.5: Convergence rate comparison of first-, second-, third- and fourth-order DCTS algorithms in random networks with different thresholds.

Fig. 2.5 shows the optimal convergence rates for the DCTS algorithms with various weights in random networks with 16 secondary users as a function of η. The results are based on 1000 realizations of the random network where we excluded disconnected networks. From the plots, we note that the first-order BC DCTS algorithm outperforms the first-order MH and MD DCTS algorithms. Furthermore, we see that the optimal convergence rate increases as M increases. However, we also observe that the fourth-order DCTS algorithm has negligible improvement compared to the third-order algorithm. Based on this, we restrict our examination of higher-order DCTS algorithm to M = 2 and 3 in the subsequent results.

In Fig. 2.6, we compare the convergence rates of the second- and third-order DCTS algorithms with the first-order DCTS algorithm for these two network topologies. Specifically, we plot the mean square time synchronization error (defined as $(1/\mathrm{N})\|\boldsymbol{\delta}(k)\|^2$). In simulating random networks, we average results over 1000 network realizations and assume $\eta = 0.9$, i.e., secondary users are well-connected with one another. We see that the second- and third-order DCTS algorithms converge faster than the first-order DCTS algorithm for both network scenarios.

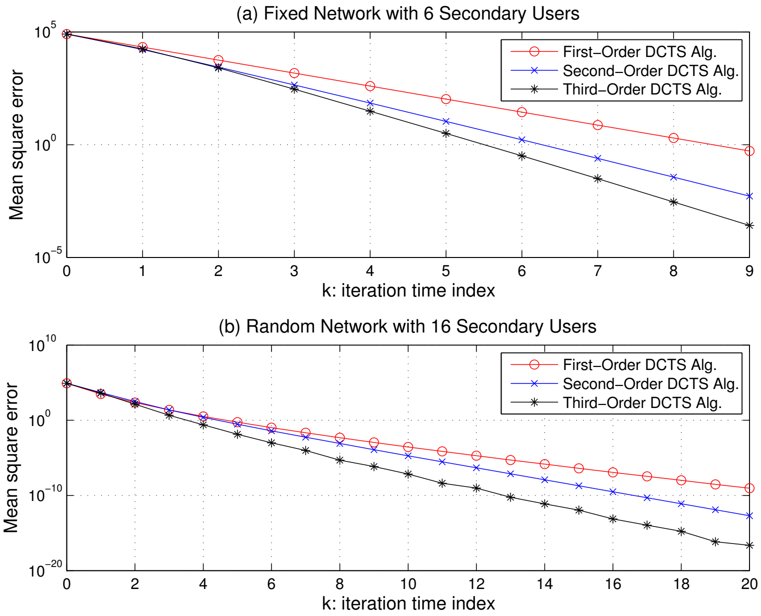


Figure 2.6: Convergence rate comparison of first-, second- and third-order DCTS algorithms: (a) Case I, (b) Case II.

(2) Convergence Properties of High-Order DCTS Algorithm

Simulation results for the evolution of the second- and third-order DCTS algorithm under Case I and Case II network topologies are shown in Fig. 2.7 and 2.8, respectively, where the results for the random network scenario are demonstrated for one (typical) network realization in Fig. 2.4(b). In the simulation, we choose the optimal values of ε and γ for the second- and third-order DCTS algorithm. As expected, as the time index increases, an average consensus is achieved when using the second- and third-order DCTS algorithms.

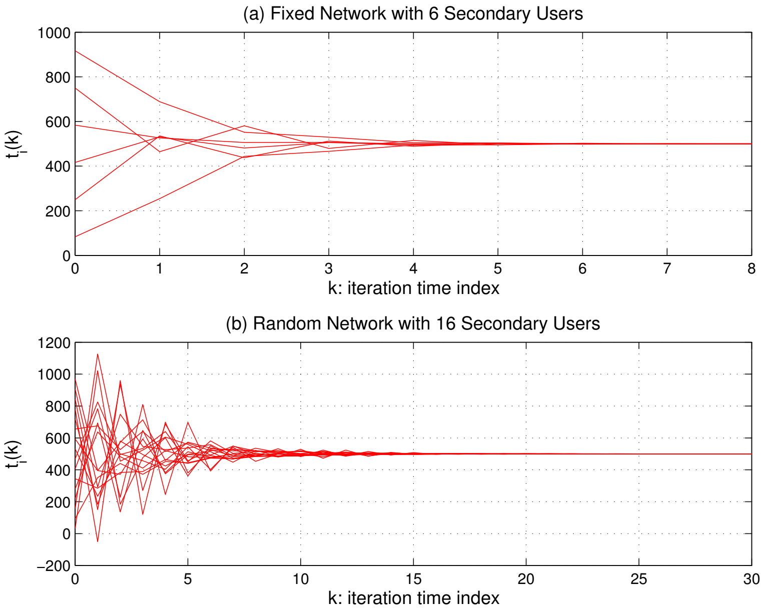


Figure 2.7: Evolution of second-order DCTS algorithm: (a) Case I, (b) Case II.

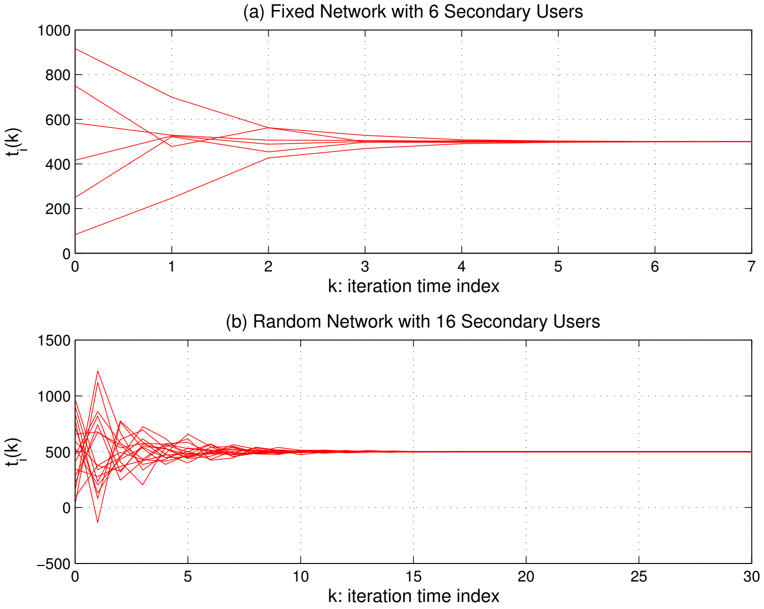


Figure 2.8: Evolution of third-order DCTS algorithm: (a) Case I, (b) Case II.

2.5 Chapter Summary

In this chapter, we have proposed a discrete time second- and high-order DCTS algorithm to address the global timing synchronization problem in ad hoc cognitive radio networks. Specifically, we have investigated the convergence region and optimal convergence rate of the second- and high-order DCTS algorithm and claimed that the optimal convergence rate of the second- and high-order DCTS algorithm is superior to that of the first-order DCTS algorithm under an appropriate algorithm design. Interestingly, the high-order discrete time consensus algorithm can be regarded as a spatial-temporal processing technique, where secondary users in the network represent the spatial advantage, the high order processing represents the temporal advantage, and the optimal convergence rate can be viewed as the diversity gain.

Chapter 3

DCTS with Gaussian Delay in Cognitive Radio Networks

3.1 Introduction

Distributed time synchronization algorithms, such as DCTS, rely heavily on local time information exchange between two or more secondary users in an ad hoc cognitive radio network. This information exchange can occur using either MAC layer time-stamped packets or via PHY layer pulse signals. Based on our knowledge, the existing body of literature on the DCTS approach does not examine the effects of time delay uncertainty between secondary users. In this chapter, we study the convergence of the DCTS algorithm when both deterministic and uncertain time delays impact local pair-wise time information exchange.

In [52], Xiao *et al.* considered distributed average consensus with additive noise and investigated the design of network link weights to minimize the mean-square deviation in steady state. Motivated by this idea, in this chapter, we analyze the convergence characteristics of the DCTS algorithm under Gaussian delay uncertainties. First, we model random delay between secondary users using a Gaussian approximation and determine the asymptotic expectation of the global synchronization error. Our results lead to the definition of a *time delay balanced network* and we claim that under such network topologies, average timing consensus between secondary users can be achieved despite the presence of random delays. Additionally, we show that the asymptotic mean square synchronization error is lower and upper bounded by several values related to network parameters. As examples, we analyze the global synchronization error of the DCTS algorithm for several structured networks.

As a secondary contribution, the analytical results presented here also extend existing literature in the broader area of distributed consensus. Specifically, earlier work [52] has examined consensus updating when the locally received information is corrupted by independent, zero-mean additive noise in each iteration of the consensus algorithm. In the timing synchronization model used here, the local information is corrupted by correlated, non-zero mean additive noise since timing uncertainties between secondary users persist in each iteration of the DCTS algorithm. As such, our results provide general convergence results for when distributed consensus is conducted with additive Gaussian corruption at each iteration.

The remainder of this chapter is outlined as follows: Section 3.2 describes time delay model for local time information exchange. Section 3.3 presents convergence results on the synchronization error of the first-order DCTS algorithm due to Gaussian random delays between secondary users. Section 3.4 discusses convergence properties of the second-order DCTS algorithm with Gaussian delays. Simulation results are presented in Section 3.5.

3.2 Time Delay Model for Local Time Information Exchange

The DCTS algorithm requires local time information exchange between two or more secondary users in a cognitive radio network. In either PHY or MAC layer based scheme, the delay between two secondary users is defined as the interval between when the time information is generated by the secondary sender and when this information is determined by the secondary receiver. Furthermore, in either case, this delay can be comprised of a deterministic and a random portion. In the following, we discuss the delay sources at the two layers and argue that in both cases, a common underlying model of Gaussian delay uncertainty[1] can be adopted.

3.2.1 PHY Layer Based Time Delay

Secondary senders using PHY layer synchronization algorithms convey local time information to secondary receivers by transmitting pulse signals according to their local clocks. The secondary receiver, however, estimates the arrival time of the

[1] We have separately examined the performance of the DCTS algorithm considering alternate delay distributions, e.g., exponential delay distribution [53]. Results show similar performance bounds as those presented in this dissertation for the Gaussian assumption. For this reason, we constrain our discussion here to the more common Gaussian delay model.

pulse signal as the clock of the secondary sender. As shown in Fig. 3.1, there is an offset between the transmit time of the pulse at the sender and the arrival time estimate at the receiver.

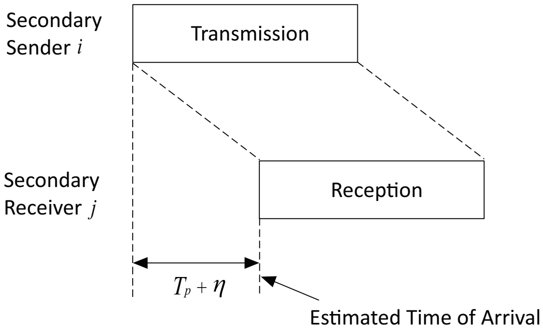


Figure 3.1: PHY layer based time delay model.

One source of this lag is T_p, the propagation delay between the secondary sender and receiver. The propagation delay is related to the distance between the two secondary users such that $T_p = \ell_{ij}/c$, where ℓ_{ij} is the distance between secondary users i and j and c is the speed of light. Once the pulse signal propagates to the receiver, the secondary receiver needs to reliably detect the pulse signal and to make an arrival time estimate. However, since the pulse signal is received in noise (may additionally experience fading over the wireless link), the actual arrival time estimate will have an associated error. It is known from parameter estimation theory that any maximum likelihood (ML) estimator is asymptotically unbiased and an ML estimate is asymptotically Gaussian distributed [54]. We assume here that an ML arrival time estimator is used and model this estimation error as a Gaussian random variable, v_{PHY}, with mean zero and variance σ^2_{PHY}. The total PHY layer delay is thus

$$T_{\mathrm{PHY}-delay} = T_p + v_{\mathrm{PHY}} \tag{3.1}$$

3.2.2 MAC Layer Based Time Delay

At the MAC layer, local time information at a secondary sender is clocked and incorporated into a packet during packet formation. The overall delay between two secondary users exchanging such time-stamped packets is, therefore, the time interval between when the sender time is clocked and when the secondary receiver decodes this time information from its received packet [16]. The sources of delay during this interval are shown in Fig. 3.2.

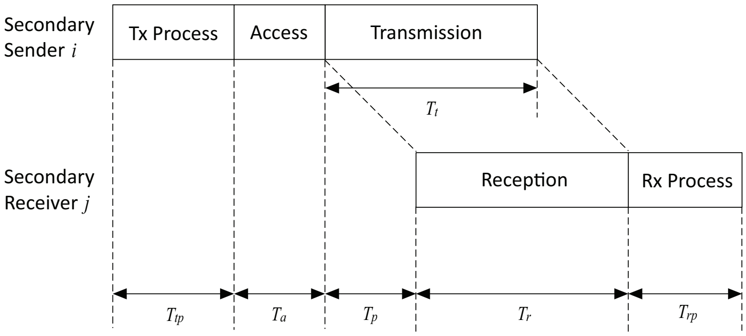


Figure 3.2: MAC layer based time delay model.

The major sources of random delay at the MAC layer are T_{tp}, the transmission processing time; T_a, the channel access time; and T_{rp}, the receiver processing time. The delay in processing a packet (at either the transmitter or receiver) depends on several factors such as the protocol processing time, the CPU load, delays in the operating system, etc.. T_a, on the other hand, is the time the secondary sender must wait to access the transmit channel, which is determined by the MAC protocol in use as well as the current network traffic. Here, we assume the overall delay, $T_{tp} + T_a + T_{rp}$ results from the additive effect of delays introduced by several independent random processes (e.g., the instantaneous workload on the sender/receiver CPU, packet generation processes at other secondary users, etc.). Using the central limit theorem, we model this delay as a Gaussian random variable with mean $\mu_{\mathrm{MAC}} = \mathbb{E}(T_{tp}) + \mathbb{E}(T_a) + \mathbb{E}(T_{rp})$ and variance $\sigma^2_{\mathrm{MAC}} = \mathbb{V}\mathrm{ar}(T_{tp}) + \mathbb{V}\mathrm{ar}(T_a) + \mathbb{V}\mathrm{ar}(T_{rp})$. Additionally, the packet experiences a propagation delay of T_p; the overall MAC layer delay is therefore given as

$$T_{\mathrm{MAC}-delay} \;=\; T_p + v_{\mathrm{MAC}}. \tag{3.2}$$

In the following, we use a general delay model that incorporates the two delay calculations for the PHY and MAC layers, i.e., we assume

$$T_{delay} \;=\; T_c + T_p + v, \tag{3.3}$$

where T_c is a constant equal to zero for PHY layer based schemes and μ_{MAC} for MAC layer based schemes; and v is a zero mean Gaussian random variable. The variance of v, σ^2, is equal to σ^2_{PHY} for PHY layer based schemes and to σ^2_{MAC} for MAC layer based schemes.

3.3 First-Order DCTS with Gaussian Delay

3.3.1 System Model

From Chapter 2, we know that in a connected, undirected network with non-random delay between secondary users, the first-order DCTS algorithm can reach *average consensus.* In this section, we focus on the operation of the first-order DCTS algorithm when there is both deterministic and random (Gaussian) delay during local time information exchange, as described above. In this case, the timing update rule of the first-order DCTS algorithm at each secondary user i is given as

$$t_i(k) \;=\; t_i(k-1) + \varepsilon \sum_{j \in \mathcal{N}_i} \left[\hat{t}_j(k-1) - t_i(k-1)\right], \tag{3.4}$$

where $\hat{t}_j(k-1) = t_j(k-1) + T_{delay} = t_j(k-1) + T_c + \ell_{ij}/c + v_j(k-1)$; T_c is the constant delay defined above and $v_j(k)$ are i.i.d Gaussian random variables, with mean zero and variance σ^2. Local time information exchange between secondary user i and j under this delay model is shown in Fig. 3.3.

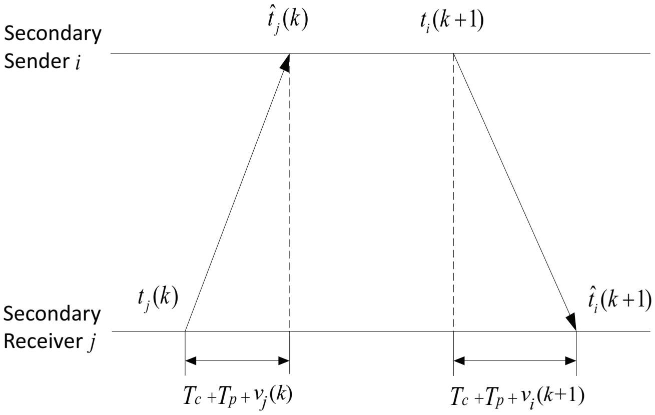


Figure 3.3: First-order DCTS algorithm with Gaussian delay during local time information exchange.

The first-order DCTS algorithm in (3.4) can be rearranged as

$$t_i(k) \;=\; t_i(k-1) + \varepsilon \sum_{j \in \mathcal{N}_i} \left[t_j(k-1) - t_i(k-1)\right] + \tilde{n}_i(k-1), \tag{3.5}$$

where $\tilde{n}_i(k-1) = \varepsilon \sum_{j \in \mathcal{N}_i} \left[T_c + \ell_{ij}/c + v_j(k-1)\right]$. It should be noted that

$\tilde{n}_i(k)$ and $\tilde{n}_j(k)$ might not be independent between secondary users i and j since the two secondary users might have identical noise coming from some potentially overlapping neighbors. Without loss of generality, in this chapter, we model the cognitive radio network as an undirected graph[2].

Let us define vector $\tilde{\boldsymbol{n}}(k) = [\tilde{n}_1(k), \tilde{n}_2(k), \cdots, \tilde{n}_{\mathrm{N}}(k)]^{\mathrm{T}}$. Then, the evolution of the first-order DCTS algorithm in (3.5) can be written as

$$\boldsymbol{t}(k) \;=\; \tilde{\mathbf{H}}\boldsymbol{t}(k-1) + \tilde{\boldsymbol{n}}(k-1), \tag{3.6}$$

where $\tilde{\mathbf{H}}$ is defined in Chapter 2. Let us define $\boldsymbol{v}(k) = [v_1(k), v_2(k), \cdots, v_{\mathrm{N}}(k)]^{\mathrm{T}}$ and $\boldsymbol{u} = [u_1, u_2, \cdots, u_{\mathrm{N}}]^{\mathrm{T}}$, where $u_i = \sum_{j\in\mathcal{N}_i}(T_c + \ell_{ij}/c)$. Then the noise vector in (3.6) is given as $\tilde{\boldsymbol{n}}(k) = \varepsilon[\boldsymbol{u} + \mathbf{A}\boldsymbol{v}(k)]$.

In the following analysis, we use the following matrices: $\mathbf{K} = (1/\mathrm{N})\mathbf{1}\mathbf{1}^{\mathrm{T}}$, $\tilde{\mathbf{P}} = \tilde{\mathbf{H}} - \mathbf{K}$ and $\tilde{\mathbf{Q}} = \mathbf{I}_{\mathrm{N}} - \mathbf{K}$. For matrices $\tilde{\mathbf{P}}$ and $\tilde{\mathbf{Q}}$, it is straightforward to show 1) the eigenvalues of $\tilde{\mathbf{P}}$ agree with those of $\tilde{\mathbf{H}}$ except that $\lambda_1(\tilde{\mathbf{H}}) = 1$ is replaced by $\lambda_1(\tilde{\mathbf{P}}) = 0$; 2) $\tilde{\mathbf{P}}^k = \tilde{\mathbf{H}}^k - \mathbf{K}$ such that $\lim\limits_{k\to\infty}\tilde{\mathbf{P}}^k = \mathbf{0}_{\mathrm{N}\times\mathrm{N}}$; and 3) $\tilde{\mathbf{Q}}\tilde{\mathbf{P}}^k\tilde{\mathbf{Q}} = \tilde{\mathbf{P}}^k$ and $\tilde{\mathbf{Q}}^k = \tilde{\mathbf{Q}}$.

3.3.2 Convergence Analysis of First-Order DCTS with Gaussian Delay

In this section, we investigate the convergence properties of the first-order DCTS algorithm with Gaussian delay. First, we derive the mean and variance of the

[2] The convergence properties presented here can be easily extended for a directed graph. We omit this extension here.

average value in each iteration of the first-order DCTS algorithm. Then, we quantify the overall impact of uncertainty by computing the first two moments of the disagreement vector, i.e., the difference between the updated times and the actual average times of the secondary users.

Recall that the average value in each iteration is defined as $m(k) = (1/\mathrm{N})\mathbf{1}^{\mathrm{T}}\boldsymbol{t}(k)$, then the mean and variance of $m(k)$ in the first-order DCTS algorithm are given in the following lemma.

Lemma 3.3.1. *For the first-order DCTS algorithm in (3.6), the mean and variance of average value $m(k)$ are given as:*

$$\begin{aligned}\mathbb{E}\left[m(k)\right] &= m(0) + \frac{k\varepsilon}{\mathrm{N}}\sum_{i=1}^{\mathrm{N}} u_i \\ \mathbb{V}\mathrm{ar}\left[m(k)\right] &= \frac{k\varepsilon^2\sigma^2}{\mathrm{N}^2}\sum_{i=1}^{\mathrm{N}} d_i^2. \end{aligned} \tag{3.7}$$

The proof of this lemma is straightforward and thus omitted from this chapter. We see that as iteration time increases, both mean and variance in (3.7) increase linearly with the time index k, i.e., as the algorithm evolves. Furthermore, the variance of $m(k)$ increases linearly with the variance of the random Gaussian delay, σ^2. Neither of these results were observed in [21], where the average value $m(k)$ was determined to be invariant during each iteration. As we will see in the following sections, although the average value $m(k)$ grows linearly with iteration time when there is Gaussian delay in the network, an average consensus may still be achievable under certain network topologies.

(1) Expectation and Second Central Moment of Disagreement Vector

Recall that the disagreement vector is defined as $\boldsymbol{\delta}(k) = \boldsymbol{t}(k) - \mathbf{K}\boldsymbol{t}(k)$. Then, the disagreement vector results in the following evolution:

$$\boldsymbol{\delta}(k) \;=\; \tilde{\mathbf{P}}\boldsymbol{\delta}(k-1) + \tilde{\mathbf{Q}}\tilde{\boldsymbol{n}}(k-1). \tag{3.8}$$

Lemma 3.3.2. *For the first-order DCTS algorithm in (3.6), the expectation of disagreement vector is given by:*

$$\mathbb{E}\left[\boldsymbol{\delta}(k)\right] \;=\; \tilde{\mathbf{P}}^k\boldsymbol{\delta}(0) + \varepsilon\sum_{l=0}^{k-1}\tilde{\mathbf{P}}^l\tilde{\mathbf{Q}}\boldsymbol{u}, \quad (k \geq 1). \tag{3.9}$$

The proof of this lemma is straightforward and thus omitted from the chapter. Let us define the second central moment of disagreement vector as $\kappa_\delta(k) \overset{\text{def}}{=} \mathbb{E}\{(\boldsymbol{\delta}(k) - \mathbb{E}[\boldsymbol{\delta}(k)])^{\mathrm{T}}(\boldsymbol{\delta}(k) - \mathbb{E}[\boldsymbol{\delta}(k)])\}$ and the covariance matrix of noise vector as $\boldsymbol{\Sigma}_n \overset{\text{def}}{=} \mathbb{E}\{(\tilde{\boldsymbol{n}}(k) - \mathbb{E}[\tilde{\boldsymbol{n}}(k)])(\tilde{\boldsymbol{n}}(k) - \mathbb{E}[\tilde{\boldsymbol{n}}(k)])^{\mathrm{T}}\} = \varepsilon^2\sigma^2\mathbf{A}^2$, where

$$\left[\mathbf{A}^2\right]_{ij} \;=\; \begin{cases} d_i & i = j \\ |\mathcal{I}_{ij}| & \text{otherwise,} \end{cases} \tag{3.10}$$

and $\mathcal{I}_{ij} = \{\mathcal{N}_i \cap \mathcal{N}_j\}$ is the set of secondary users that are neighbors of both secondary user i and j. Given these definitions, we next note that

Lemma 3.3.3. *For the first-order DCTS algorithm in (3.6), the second central moment of disagreement vector is given as:*

$$\kappa_\delta(k) \;=\; \boldsymbol{\delta}(0)^{\mathrm{T}}\tilde{\mathbf{P}}^{2k}\boldsymbol{\delta}(0) + \varepsilon^2\sigma^2\mathbf{tr}\left[\tilde{\mathbf{Q}}\sum_{l=0}^{k-1}\tilde{\mathbf{P}}^{2l}\tilde{\mathbf{Q}}\mathbf{A}^2\right], \quad (k \geq 1). \tag{3.11}$$

Proof. Please see the Appendix B.1. □

(2) Asymptotic Expectation of Global Synchronization Error

Using Lemma 3.3.2, we see that the steady state of expectation of disagreement vector is

$$\boldsymbol{\mu}_\infty \overset{\text{def}}{=} \lim_{k\to\infty} \mathbb{E}\left[\boldsymbol{\delta}(k)\right] = \varepsilon \mathbf{W}_1 \tilde{\mathbf{Q}} \boldsymbol{u}, \tag{3.12}$$

where $\mathbf{W}_1 = \left(\mathbf{I}_\text{N} - \tilde{\mathbf{P}}\right)^{-1}$. The equation holds because $\lim_{k\to\infty} \tilde{\mathbf{P}}^k = 0$. It is easy to check that the eigenvalues of $\mathbf{W}_1$ are $\lambda_1(\mathbf{W}_1) = 1$ and $\lambda_i(\mathbf{W}_1) = 1/[\varepsilon\lambda_i(\mathbf{L})],\ i = 2, \cdots, \text{N}$. For this $\boldsymbol{\mu}_\infty$, we can show that

Theorem 3.3.1. *In a network with fixed, connected topology, $\boldsymbol{\mu}_\infty$ in (3.12) is a constant vector independent of the constant value of ε.*

Proof. Let us denote the eigenvectors of $\mathbf{W}_1$ as $\boldsymbol{\omega}_i$. It is easy to check that the eigenvector corresponding to $\lambda_1(\mathbf{W}_1) = 1$ is $\boldsymbol{\omega}_1 = \mathbf{1}$. $\boldsymbol{\mu}_\infty$ in (3.12) can thus be written as

$$\begin{aligned}
\boldsymbol{\mu}_\infty &= \varepsilon \mathbf{1}\mathbf{1}^\mathsf{T} \tilde{\mathbf{Q}} \boldsymbol{u} + \left[\sum_{i=2}^{\text{N}} \frac{1}{\varepsilon\lambda_i(\mathbf{L})} \boldsymbol{\omega}_i \boldsymbol{\omega}_i^\mathsf{T}\right] \tilde{\mathbf{Q}} \varepsilon \boldsymbol{u} \\
&= \left[\sum_{i=2}^{\text{N}} \frac{1}{\lambda_i(\mathbf{L})} \boldsymbol{\omega}_i \boldsymbol{\omega}_i^\mathsf{T}\right] \tilde{\mathbf{Q}} \boldsymbol{u} \\
&= \left(\mathbf{L} + \mathbf{K}\right)^{-1} \tilde{\mathbf{Q}} \boldsymbol{u}.
\end{aligned} \tag{3.13}$$

Thus, $\boldsymbol{\mu}_\infty$ does not depend on ε. This completes the proof. □

Thus, for a constant step size ε, the steady state of expectation of disagreement vector is a constant vector regardless of ε. In other words, in a network with fixed topology, the expectation of global synchronization error is the same regardless of the speed of synchronization.

In general, we see that the first-order DCTS algorithm with Gaussian delay cannot achieve average consensus since $\boldsymbol{\mu}_\infty$ is a linear function of $\boldsymbol{u}$ (is not equal to $\mathbf{0}$). This global synchronization error can be viewed as the accuracy of time synchronization algorithm. If this synchronization error is tolerable or small compared to time resolution of the system, we say that this first-order DCTS algorithm still achieves the average consensus, but with *"tolerable synchronization error"*. Let us now define the asymptotic expectation of pair-wise synchronization error as

$$\Delta t_{i,j} \;=\; \lim_{k\to\infty} \mathbb{E}\left[t_i(k) - t_j(k)\right] \;=\; \mu_{i,\infty} - \mu_{j,\infty}, \quad i, j \in \mathcal{V}.$$

Hence, the maximum asymptotic expectation of global synchronization error between any two secondary users is

$$\Delta t_{\max} \;=\; \max\left\{|\Delta t_{i,j}|\right\}, \quad i, j \in \mathcal{V}. \tag{3.14}$$

Definition 3.3.1. *A connected network is called "average consensus achievable with tolerable synchronization error" if the maximum asymptotic expectation of global time synchronization error in (3.14) is less than a predefined threshold Δt_{Th} when applying the first-order DCTS algorithm in (3.6), i.e., when $\Delta t_{\max} < \Delta t_{Th}$.*

It is worth mentioning that, under certain network topologies, we can still asymptotically achieve the average consensus when utilizing the first-order DCTS algorithm with Gaussian delay. Recall that $\boldsymbol{\mu}_\infty = (\mathbf{L}+\mathbf{K})^{-1}\tilde{\mathbf{Q}}\boldsymbol{u}$. In this equation, $\tilde{\mathbf{Q}}\boldsymbol{u} = \boldsymbol{u} - \mathbf{K}\boldsymbol{u}$ is the disagreement vector of $\boldsymbol{u}$. When $\boldsymbol{u} = \mathbf{K}\boldsymbol{u}$, we see that

$$\sum_{j\in\mathcal{N}_i} (T_c + \ell_{ij}/c) \;=\; \sum_{m\in\mathcal{N}_k} (T_c + \ell_{km}/c)\,,\ (i,j) \in \mathcal{E} \text{ and } (k,m) \in \mathcal{E}.$$

More specifically, when

$$d_i \;=\; d_j, \quad (i,j \in \mathcal{V}) \quad \text{and} \quad \ell_{km} \;=\; \ell_{lr}, \quad (k,m) \in \mathcal{E},\ (l,r) \in \mathcal{E},$$

then $\boldsymbol{\mu}_\infty = \mathbf{0}$ and $\Delta t_{\max} = 0$, implying that the first-order DCTS algorithm achieves average consensus asymptotically. The condition above indicates that the time delay between secondary users can be canceled if each secondary user receives the same amount of time delay from all neighbor secondary users. Networks in this condition can be met are defined in the following:

Definition 3.3.2. *A network is called "time delay balanced network" if the delay*

$$\sum_{j\in\mathcal{N}_i} (T_c + \ell_{ij}/c) \;=\; \sum_{m\in\mathcal{N}_k} (T_c + \ell_{km}/c)\,, \quad (i,j) \in \mathcal{E} \textit{ and } (k,m) \in \mathcal{E},$$

or equivalently, $\Delta t_{\max} = 0$.

Otherwise we refer to the network as "*time delay unbalanced*". It is worth mentioning that a similar definition of "*equal delay networks*" was discussed in [55] for continuous time network synchronization. Based on the definition above, we see that time delay balance may be readily (but not exclusively) achieved in well-structured networks.

(3) Asymptotic Mean Square Time Synchronization Error

Using Lemma 3.3.3, the steady state of second central moment of disagreement vector is

$$\kappa_{\delta,\infty} \overset{\text{def}}{=} \lim_{k\to\infty} \kappa_\delta(k) \; = \; \varepsilon^2\sigma^2 \mathbf{tr}\left(\mathbf{W}_2\mathbf{A}^2\right), \tag{3.15}$$

where $\mathbf{W}_2 = (\mathbf{I}_{\mathrm{N}} - \tilde{\mathbf{P}}^2)^{-1} + \tilde{\mathbf{Q}} - \mathbf{I}_{\mathrm{N}}$. Then, the eigenvalues of $\mathbf{W}_2$ are $\lambda_1(\mathbf{W}_2) = 0$ and $\lambda_i(\mathbf{W}_2) = 1/[2\varepsilon\lambda_i(\mathbf{L}) - \varepsilon^2\lambda_i^2(\mathbf{L})]$, $i = 2, \cdots, \mathrm{N}$.

Let us define the asymptotic mean square time synchronization error as

$$\sigma_{\Delta t}^2 \overset{\text{def}}{=} \lim_{k\to\infty} \sum_{i=1}^{\mathrm{N}} \mathbb{E}\big[\, |t_i(k) - m(k)|^2 \,\big], \tag{3.16}$$

which indicates the amount of error by which the updated time at each secondary user differs from the average value over all N secondary users. In particular, we see that

$$\sigma_{\Delta t}^2 \; = \; \boldsymbol{u}^{\mathrm{T}}\tilde{\mathbf{Q}}\left(\mathbf{L} + \mathbf{K}\right)^{-2}\tilde{\mathbf{Q}}\boldsymbol{u} + \varepsilon^2\sigma^2\mathbf{tr}\left(\mathbf{W}_2\mathbf{A}^2\right). \tag{3.17}$$

Theorem 3.3.2. *For a connected, time delay unbalanced network, $\sigma_{\Delta t}^2$ in (3.16) is bounded by*

$$\begin{aligned} \sigma_{\Delta t}^2 \; &\geq \; \frac{\boldsymbol{u}^{\mathrm{T}}\tilde{\mathbf{Q}}\boldsymbol{u}}{\alpha_1} + \varepsilon\sigma^2\lambda_{\min}(\mathbf{A}^2)\sum_{i=2}^{\mathrm{N}}\lambda_i \\ \sigma_{\Delta t}^2 \; &\leq \; \frac{\|\boldsymbol{u}\|^2}{\alpha_2} + \varepsilon\sigma^2\min\left\{\mathcal{D}_{\mathrm{N}}\max\left\{\lambda_i\right\}, \lambda_{\max}(\mathbf{A}^2)\sum_{i=2}^{\mathrm{N}}\lambda_i\right\}, \end{aligned} \tag{3.18}$$

where $\alpha_1 = \lambda_{\mathrm{N}}^2(\mathbf{L})$, $\alpha_2 = \min\{\lambda_2^2(\mathbf{L}), 1\}$, $\lambda_i = 1/[2\lambda_i(\mathbf{L}) - \varepsilon\lambda_i^2(\mathbf{L})]$, $i = 2, \cdots, \mathrm{N}$ and $\mathcal{D}_{\mathrm{N}} = \sum_{i=1}^{\mathrm{N}} d_i$ is the total degree in the networks.

Proof. Please see the Appendix B.2. □

Hence, we see that the lower and upper bounds of $\sigma^2_{\Delta t}$ are determined by several values related to network parameters: eigenvalues of $\mathbf{L}$ and $\mathbf{A}^2$, total degree of network, step size and delay time vector.

3.3.3 First-Order DCTS with Gaussian Delay in Structured Networks

In this section, we apply the first-order DCTS algorithm under Gaussian delay for several structured networks. We note that structured networks studied here may not necessarily be feasible in typical cognitive radio network deployments; we study them as they are analytically tractable, provide some valuable insights and can be used to validate our analytical findings above. Specifically, we study the following network topologies:

Definition 3.3.3. *" A Complete Network with Equal Distance Degree (K_{N})": In a complete network, every pair of distinct secondary users is connected by an edge. A complete network with equal distance degree is a complete network that has* N *secondary users, edge set $\{(i,j), i \neq j\}$ and $\sum_{j\in\mathcal{N}_i} \ell_{ij} = \sum_{i\in\mathcal{N}_j} \ell_{ji}$.*

Definition 3.3.4. *" A Ring Network with Equal Distance (R_{N})": A ring network is a network that consists of a single cycle. The ring network with equal distance is a ring network that has* N *secondary users,* N *edges and $\ell_c = \ell_{ij} = \ell_{km}$ for $(i,j) \in \mathcal{E}$ and $(k,m) \in \mathcal{E}$.*

Definition 3.3.5. *" A Path Network with Equal Distance (P_{N})": A path network* *is a network that consists of edge set* $\{(i, i+1), 1 \leq i < \mathrm{N}\}$*. The path network with equal distance is a path network that has* N *secondary users,* $\mathrm{N}-1$ *edges and* $\ell_c = \ell_{ij} = \ell_{km}$ *for* $(i,j) \in \mathcal{E}$ *and* $(k,m) \in \mathcal{E}$*.*

Definition 3.3.6. *" A Star Network with Equal Distance* (S_N)*": A star network is a network that consists of edge set* $\{(i, \mathrm{N}), 1 \leq i < \mathrm{N}\}$*. The star network with equal distance is a star network that has* N *secondary users,* $\mathrm{N}-1$ *edges and* $\ell_c = \ell_{ij} = \ell_{km}$ *for* $(i,j) \in \mathcal{E}$ *and* $(k,m) \in \mathcal{E}$*.*

Definition 3.3.7. *" A Hypercube Network with Equal Distance Degree* (H_N)*":A hypercube network with equal distance degree is a hypercube network that has* N *secondary users,* $\mathrm{N}\log_2 \mathrm{N}$ *edges and* $\sum_{j \in \mathcal{N}_i} \ell_{ij} = \sum_{i \in \mathcal{N}_j} \ell_{ji}$*.*

Fig. 3.4 shows the examples of structured networks: a complete network K_5, a ring network R_8, a path network P_5, a star network S_8 and a hypercube network H_8. We now explore the convergence properties of the global synchronization error for these structured networks.

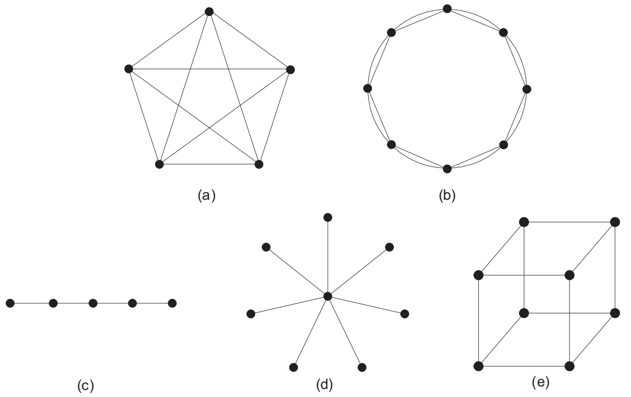


Figure 3.4: Structured networks: (a) K_5, (b) R_8, (c) P_5, (d) S_8, (e) H_8.

(1) Convergence Properties for Complete Networks

The convergence characteristic of global synchronization error for a complete network is described in the following lemma.

Lemma 3.3.4. *For a complete network with* N *secondary users and equal distance degree, the first-order DCTS algorithm in (3.6) yields a global synchronization error with the following properties:*

$$\begin{aligned} \Delta t_{\max} &= 0 \\ \sigma^2_{\Delta t} &= \frac{(\mathrm{N}-1)\varepsilon\sigma^2}{2\mathrm{N}-\varepsilon\mathrm{N}^2}. \end{aligned} \tag{3.19}$$

Proof. The first equation holds because K_{N} has the following property as mentioned earlier:

$$\tilde{\mathbf{Q}}\boldsymbol{u} = \mathbf{0} \Rightarrow \boldsymbol{\mu}_{\infty} = \mathbf{0} \Rightarrow \Delta t_{\max} = 0.$$

Now let us prove the second property. Recall that for a complete network, Laplacian matrix $\mathbf{L}$ has eigenvalue 0 with multiplicity 1 and N with multiplicity $\mathrm{N}-1$, i.e., $\lambda_1(\mathbf{L}) = 0$ and $\lambda_i(\mathbf{L}) = \mathrm{N}, \quad i = 2, 3, \cdots, \mathrm{N}$, and the total number of degree is $\mathcal{D}_{\mathrm{N}} = \mathrm{N}(\mathrm{N}-1)$. Furthermore, we have

$$\left[\mathbf{A}^2\right]_{ij} = \begin{cases} \mathrm{N}-1 & i = j \\ \mathrm{N}-2 & \text{otherwise.} \end{cases} \tag{3.20}$$

In addition, $\mathbf{W}_2$ can be simplified as $1/(2\varepsilon\mathrm{N} - \varepsilon^2\mathrm{N}^2)\tilde{\mathbf{Q}}$.

Plugging (3.20) and $\mathbf{W}_2$ into (3.17) and keeping in mind that $\tilde{\mathbf{Q}}\boldsymbol{u} = \mathbf{0}$, then we have

$$\sigma_{\Delta t}^2 \;=\; \frac{\varepsilon\sigma^2}{2\mathrm{N}-\varepsilon\mathrm{N}^2}\left[\mathcal{D}_{\mathrm{N}}-(\mathrm{N}-1)^2\right] \;=\; \frac{(\mathrm{N}-1)\varepsilon\sigma^2}{2\mathrm{N}-\varepsilon\mathrm{N}^2}.$$

This completes the proof. □

Thus, the complete network K_{N} is a time delay balanced network. It should be noted that when the first-order DCTS algorithm is employed with the optimal step size, we get that $\sigma_{\Delta t}^2 = (\mathrm{N}-1)\sigma^2/\mathrm{N}^2$ which is a decreasing function of N. Furthermore, as N becomes large, $\sigma_{\Delta t}^2 \approx \sigma^2/\mathrm{N}$.

(2) Convergence Properties for Ring Networks

For a ring network, Laplacian matrix $\mathbf{L}$ is a circulant matrix and the eigenvalues of $\mathbf{L}$ can be calculated analytically as [56]: $\lambda_i(\mathbf{L}) = 2 - 2\cos\left[2\pi(i-1)/\mathrm{N}\right],\ i = 1, 2, \cdots, \mathrm{N}$. Total degree for a ring network is $\mathcal{D}_{\mathrm{N}} = 2\mathrm{N}$. Moreover, $\mathbf{A}^2$ is a circulant matrix with eigenvalues $\lambda_i(\mathbf{A}^2) = 2\left\{1+\cos\left[4\pi(i-1)/\mathrm{N}\right]\right\},\ i = 1, 2, \cdots, \mathrm{N}$. It is easy to show that $\lambda_{\max}(\mathbf{A}^2) = \lambda_1(\mathbf{A}^2)$ and $\lambda_{\min}(\mathbf{A}^2) = \lambda_{\lfloor(\mathrm{N}+5)/4\rfloor}(\mathbf{A}^2)$.

According to Theorem 3.3.2, we see that for a ring network with N secondary users and equal distance, the first-order DCTS algorithm in (3.6) produces a global synchronization error with the following properties:

$$\begin{aligned}
\Delta t_{\max} &= 0 \\
\sigma_{\Delta t}^2 &\geq \frac{\varepsilon\sigma^2}{2}\left[1+\cos\left(\frac{4\pi\lfloor(\mathrm{N}+1)/4\rfloor}{\mathrm{N}}\right)\right]\sum_{i=1}^{\mathrm{N}-1}\lambda_i \\
\sigma_{\Delta t}^2 &\leq \varepsilon\sigma^2\min\left\{\max\left\{\frac{\mathrm{N}\lambda_i}{2}\right\}, \sum_{i=1}^{\mathrm{N}-1}\lambda_i\right\},
\end{aligned} \tag{3.21}$$

where $\lambda_i = 1/[1-\varepsilon+(2\varepsilon-1)\cos(2\pi i/\mathrm{N})-\varepsilon\cos^2(2\pi i/\mathrm{N})], i=1,\cdots,\mathrm{N}-1$. Thus the ring network R_N studied here is also a time delay balanced network.

(3) Convergence Properties for Path Networks

For a path network, the eigenvalues of $\mathbf{L}$ are equivalent to those of $R_{2\mathrm{N}}$ [57] and can be calculated analytically as [49]: $\lambda_i(\mathbf{L}) = 2-2\cos[\pi(i-1)/\mathrm{N}]\,,\ i=1,2,\cdots,\mathrm{N}$. It should be noted that $\lambda_\mathrm{N}(\mathbf{L}) = 2-2\cos[\pi(\mathrm{N}-1)/\mathrm{N}]$, $\lambda_2(\mathbf{L}) = 2-2\cos(\pi/\mathrm{N})$ and when $\mathrm{N}>2$, $\lambda_2(\mathbf{L})<1$ and $\lambda_\mathrm{N}(\mathbf{L})>1$. Also, total degree for a path network is $\mathcal{D}_\mathrm{N} = 2(\mathrm{N}-1)$. Furthermore, the eigenvalues of $\mathbf{A}^2$ are $\lambda_i(\mathbf{A}^2) = 2\left[1+\cos(2\pi i/(\mathrm{N}+1))\right],\ i=1,2,\cdots,\mathrm{N}$. It is easy to check that $\lambda_{\max}(\mathbf{A}^2) = \lambda_1(\mathbf{A}^2)$ and $\lambda_{\min}(\mathbf{A}^2) = \lambda_{\lfloor(\mathrm{N}+1)/2\rfloor}(\mathbf{A}^2)$.

According to Theorem 3.3.2, we see that for a path network with N secondary users and equal distance P_N, the first-order DCTS algorithm in (3.6) yields a global synchronization error with the following properties:

$$\begin{aligned}
\sigma^2_{\Delta t} &\geq \frac{\boldsymbol{u}^\mathrm{T}\tilde{\mathbf{Q}}\boldsymbol{u}}{\alpha_1} + \frac{\varepsilon\sigma^2}{2}\left[1+\cos\left(\frac{2\pi\lfloor(\mathrm{N}+1)/2\rfloor}{\mathrm{N}+1}\right)\right]\sum_{i=1}^{\mathrm{N}-1}\lambda_i \\
\sigma^2_{\Delta t} &\leq \frac{\|\boldsymbol{u}\|^2}{\alpha_2} + \frac{\varepsilon\sigma^2}{2}\min\left\{\max\{(\mathrm{N}-1)\lambda_i\},\left[1+\cos\left(\frac{2\pi}{\mathrm{N}+1}\right)\right]\sum_{i=1}^{\mathrm{N}-1}\lambda_i\right\}, \quad (3.22)
\end{aligned}$$

where $\alpha_1 = 4\left[1-\cos(\pi(\mathrm{N}-1)/\mathrm{N})\right]^2$, $\alpha_2 = 4\left[1-\cos(\pi/\mathrm{N})\right]^2$ and $\lambda_i = 1/[1-\varepsilon+(2\varepsilon-1)\cos(\pi i/\mathrm{N})-\varepsilon\cos^2(\pi i/\mathrm{N})],\ i=1,\cdots,\mathrm{N}-1,\ (\mathrm{N}>2)$.

It is worth mentioning that for a path network, $\Delta t_{\max}\neq 0$ because the degrees of secondary user 1 and N are different from other secondary users. Let us define $u_{cp} = T_c + \ell_c/c$. As N becomes large, we have $\boldsymbol{\mu}_\infty \approx -u_{cp}\left(\mathbf{L}+\mathbf{K}\right)^{-1}(e_1+e_\mathrm{N})$, where $e_i = [0 \; \cdots \; 0 \; 1 \; 0 \; \cdots \; 0]^{\mathrm{T}}$ is an $\mathrm{N} \times 1$ unit vector with the ith component equal to 1.

(4) Convergence Properties for Star Networks

For a star network, the eigenvalues of $\mathbf{L}$ are $\lambda_1(\mathbf{L}) = 0,\ \lambda_{\mathrm{N}}(\mathbf{L}) = \mathrm{N}$ and $\lambda_i(\mathbf{L}) = 1,\ i = 2, \cdots, \mathrm{N}-1$. Total degree for a star network is $\mathcal{D}_{\mathrm{N}} = 2(\mathrm{N}-1)$. Moreover, the eigenvalues of $\mathbf{A}^2$ are $\lambda_{\mathrm{N}-1}(\mathbf{A}^2) = \lambda_{\mathrm{N}}(\mathbf{A}^2) = \mathrm{N}-1$ and $\lambda_i(\mathbf{A}^2) = 0, \quad i = 1, \cdots, \mathrm{N}-2$. Hence, $\lambda_{\max}(\mathbf{A}^2) = \mathrm{N}-1$ and $\lambda_{\min}(\mathbf{A}^2) = 0$.

According to Theorem 3.3.2, we see that for a star network with N secondary users and equal distance S_{N}, the first-order DCTS algorithm in (3.6) produces a global synchronization error with the following properties:

$$\begin{aligned} \sigma^2_{\Delta t} &\geq \frac{\boldsymbol{u}^{\mathrm{T}}\tilde{\mathbf{Q}}\boldsymbol{u}}{\mathrm{N}^2} \\ \sigma^2_{\Delta t} &\leq \|\boldsymbol{u}\|^2 + (\mathrm{N}-1)\varepsilon\sigma^2 \cdot \min\left\{\max\left\{2\lambda_1, 2\lambda_2\right\}, (\mathrm{N}-2)\lambda_1 + \lambda_2\right\}, \end{aligned} \tag{3.23}$$

where $\lambda_1 = 1/(2-\varepsilon)$ and $\lambda_2 = 1/(2\mathrm{N} - \varepsilon\mathrm{N}^2)$.

It should be noted that when operating the first-order DCTS algorithm with an optimal constant step size $\varepsilon_{\text{opt, FO}}$, the steady state of second central moment of disagreement vector is $\kappa_{\delta,\infty} = (\mathrm{N}-1)\sigma^2/\mathrm{N}$. This is because $\mathbf{W}_2$ can be simplified in this case to $\frac{(\mathrm{N}+1)^2}{4\mathrm{N}}\tilde{\mathbf{Q}}$. As a result, we see that as N becomes large, $\kappa_{\delta,\infty} \approx \sigma^2$. Furthermore, just in the case of the path network, as N becomes large, we have $\boldsymbol{\mu}_\infty \approx u_{cp}\left(\mathbf{L} + \mathbf{K}\right)^{-1}\left(-\sum_{i=1}^{\mathrm{N}-1} e_i + \mathrm{N}e_{\mathrm{N}}\right)$.

(5) Convergence Properties for Hypercube Networks

For a hypercube network, the eigenvalues of $\mathbf{L}$ are $2i,\ i = 0, 1, \cdots, \mathcal{Z}_\mathrm{N}$, each with multiplicity $\binom{\mathrm{N}}{i}$ [58], where $\mathcal{Z}_\mathrm{N} = \log_2 \mathrm{N}$. Total degree for a hypercube network is $\mathcal{D}_\mathrm{N} = \mathrm{N}\mathcal{Z}_\mathrm{N}$. Moreover, the eigenvalues of $\mathbf{A}^2$ are obtained as follows: when $\mathcal{Z}_\mathrm{N}$ is even, $\lambda_i(\mathbf{A}^2) = 4i^2,\ i = 0, 1, \cdots, \mathcal{Z}_\mathrm{N}/2$, and 0 with multiplicity $\binom{\mathcal{Z}_\mathrm{N}}{\mathcal{Z}_\mathrm{N}/2}$, others with multiplicity $2\binom{\mathcal{Z}_\mathrm{N}}{\mathcal{Z}_\mathrm{N}/2+i}$, $i = 1, 2, \cdots, \mathcal{Z}_\mathrm{N}/2$; when $\mathcal{Z}_\mathrm{N}$ is odd, $\lambda_i(\mathbf{A}^2) = (2i+1)^2,\ i = 0, 1, \cdots, \lfloor \mathcal{Z}_\mathrm{N}/2 \rfloor$, each with multiplicity $2\binom{\mathcal{Z}_\mathrm{N}}{\lfloor \mathcal{Z}_\mathrm{N}/2 \rfloor + i + 1}$. It is easy to show that $\lambda_{\max}(\mathbf{A}^2) = \mathcal{Z}_\mathrm{N}^2$, $\lambda_{\min}(\mathbf{A}^2) = 0$ when $\mathcal{Z}_\mathrm{N}$ is even and $\lambda_{\min}(\mathbf{A}^2) = 1$ when $\mathcal{Z}_\mathrm{N}$ is odd.

According to Theorem 3.3.2, we see that for a hypercube network with N secondary users and equal distance degree, the first-order DCTS algorithm in (3.6) yields a global synchronization error with the following properties:

$$\begin{aligned}
\Delta t_{\max} &= 0 \\
\sigma^2_{\Delta t} &\geq \begin{cases} 0 & \text{when } \mathcal{Z}_\mathrm{N} \text{ is even} \\ \varepsilon\sigma^2 \sum_{i=1}^{\mathcal{Z}_\mathrm{N}} \binom{\mathcal{Z}_\mathrm{N}}{i} \lambda_i & \text{when } \mathcal{Z}_\mathrm{N} \text{ is odd} \end{cases} \\
\sigma^2_{\Delta t} &\leq \varepsilon\sigma^2 \mathcal{Z}_\mathrm{N} \min\left\{ \max\{\mathrm{N}\lambda_i\}, \mathcal{Z}_\mathrm{N} \sum_{i=1}^{\mathcal{Z}_\mathrm{N}} \binom{\mathcal{Z}_\mathrm{N}}{i} \lambda_i \right\},
\end{aligned} \tag{3.24}$$

where $\lambda_i = 1/(4i - 4\varepsilon i^2),\ i = 1, \cdots, \mathcal{Z}_\mathrm{N}$. Since $\Delta t_{\max} = 0$, the hypercube network is also time delay balanced.

3.4 Second-Order DCTS with Gaussian Delay

3.4.1 System Model

In this section, we investigate the convergence properties of the second-order DCTS algorithm in undirected networks with Gaussian delay. Then, the timing update rule of the second-order DCTS algorithm at each secondary user i is given as

$$\begin{aligned} t_i(k) &= t_i(k-1) + \varepsilon \sum_{j \in \mathcal{N}_i} \left[\hat{t}_j(k-1) - t_i(k-1)\right] \\ &- \gamma\varepsilon \sum_{j \in \mathcal{N}_i} \left[\hat{t}_j(k-2) - t_i(k-2)\right], \end{aligned} \tag{3.25}$$

Similar to the first-order DCTS algorithm and according to the matrix theory, the evolution of the second-order DCTS algorithm in (3.25) can be written as

$$\boldsymbol{t}(k) = (\mathbf{I}_{\mathrm{N}} - \varepsilon\mathbf{L})\,\boldsymbol{t}(k-1) + \gamma\varepsilon\mathbf{L}\boldsymbol{t}(k-2) + \boldsymbol{n}(k-1), \tag{3.26}$$

where $\boldsymbol{n}(k-1) = \varepsilon(1-\gamma)\boldsymbol{u} + \varepsilon\mathbf{A}(\boldsymbol{v}(k-1) - \gamma\boldsymbol{v}(k-2))$.

Recall that $\boldsymbol{\psi}(k) = [\boldsymbol{t}(k)^{\mathrm{T}}\ \boldsymbol{t}(k-1)^{\mathrm{T}}]^{\mathrm{T}}$. Let us additionally define $\boldsymbol{\zeta}(k) = [\boldsymbol{n}(k)^{\mathrm{T}}\ \mathbf{0}^{\mathrm{T}}]^{\mathrm{T}}$ and

$$\mathbf{H} = \begin{bmatrix} \mathbf{I}_{\mathrm{N}} - \varepsilon\mathbf{L} & \gamma\varepsilon\mathbf{L} \\ \mathbf{I}_{\mathrm{N}} & \mathbf{0}_{\mathrm{N}\times\mathrm{N}} \end{bmatrix}.$$

Then, (3.26) can be rewritten as

$$\boldsymbol{\psi}(k) = \mathbf{H}\boldsymbol{\psi}(k-1) + \boldsymbol{\zeta}(k-1). \tag{3.27}$$

The mean and variance of the average value $m(k)$ are given in the following lemma.

Lemma 3.4.1. *For the second-order DCTS algorithm in (3.26), the mean and variance of the average value of* $m(k)$ *are given as:*

$$\begin{aligned} \mathbb{E}\left[m(k)\right] &= m(0) + \frac{k}{\mathrm{N}}\sum_{i=1}^{\mathrm{N}} u_i \\ \mathbb{V}\mathrm{ar}\left[m(k)\right] &= \frac{k\varepsilon^2\sigma^2(1+\gamma^2)}{\mathrm{N}^2}\sum_{i=1}^{\mathrm{N}} d_i^2. \end{aligned} \tag{3.28}$$

The proof of this lemma is straightforward and thus omitted from this chapter. Similar to the first-order DCTS algorithm, we see that as iteration time increases, both mean and variance in (3.28) increase linearly with the time index k. Furthermore, the variance of $m(k)$ increases linearly with σ^2. As we will see in our numerical results, although the average value $m(k)$ grows linearly with iteration time when there is Gaussian delay in the network, an average consensus may still be achievable under certain network topologies.

3.4.2 Convergence Analysis of Second-Order DCTS with Gaussian Delay

(1) Expectation and Second Central Moment of Error Vector

In order to understand the convergence property of second-order DCTS algorithm with Gaussian delay, we first quantify the overall impact of uncertainty by computing the first two moments of the disagreement vector. Recall that

$\boldsymbol{e}(k) = \left[\boldsymbol{\delta}^{\mathrm{T}}(k)\ \boldsymbol{\delta}^{\mathrm{T}}(k-1)\right]^{\mathrm{T}}$. With Gaussian delay, we see that the error vector $\boldsymbol{e}(k)$ results in the following evolution:

$$\boldsymbol{e}(k) \;=\; \mathbf{P}\boldsymbol{e}(k-1) + \mathbf{Q}\boldsymbol{\zeta}(k-1), \tag{3.29}$$

where $\mathbf{P} = \mathbf{H} - \mathbf{J}$, $\mathbf{Q} = \mathbf{I}_{2\mathrm{N}} - \mathbf{J}_1$. Here

$$\mathbf{J} \;=\; \begin{bmatrix} \mathbf{K} & \mathbf{0}_{\mathrm{N}\times\mathrm{N}} \\ \mathbf{K} & \mathbf{0}_{\mathrm{N}\times\mathrm{N}} \end{bmatrix} \text{ and } \mathbf{J}_1 \;=\; \begin{bmatrix} \mathbf{K} & \mathbf{0}_{\mathrm{N}\times\mathrm{N}} \\ \mathbf{0}_{\mathrm{N}\times\mathrm{N}} & \mathbf{K} \end{bmatrix}.$$

Then, we have

Lemma 3.4.2. *For the second-order DCTS algorithm in (3.26), the expectation of the error vector $\boldsymbol{e}(k)$ is given by*

$$\boldsymbol{e}(k) \;=\; \mathbf{P}^k\boldsymbol{e}(0) + (1-\gamma)\varepsilon\sum_{l=0}^{k-1}\mathbf{P}^l\mathbf{Q}\boldsymbol{u}_1, \quad (k \geq 1),$$

where $\boldsymbol{u}_1 = [\boldsymbol{u}^{\mathrm{T}}\ \mathbf{0}^{\mathrm{T}}]^{\mathrm{T}}$.

The proof of this lemma is straightforward and thus omitted from this chapter. Let us define the second central moment of the error vector as $\kappa_e(k) \overset{\mathrm{def}}{=} \mathbb{E}\{(\boldsymbol{e}(k) - \mathbb{E}[\boldsymbol{e}(k)])^{\mathrm{T}}(\boldsymbol{e}(k) - \mathbb{E}[\boldsymbol{e}(k)])\}$ and the covariance matrix of the error vector as $\boldsymbol{\Sigma}_e(k) \overset{\mathrm{def}}{=} \mathbb{E}\{(\boldsymbol{e}(k) - \mathbb{E}[\boldsymbol{e}(k)])(\boldsymbol{e}(k) - \mathbb{E}[\boldsymbol{e}(k)])^{\mathrm{T}}\}$. It is worth mentioning that $\kappa_e(k) = \mathbf{tr}(\boldsymbol{\Sigma}_e(k))$. Additionally, let us denote the covariance matrix of $\boldsymbol{\zeta}(k)$ as $\boldsymbol{\Sigma}_\zeta \overset{\mathrm{def}}{=} \mathbb{E}\{(\boldsymbol{\zeta}(k) - \mathbb{E}[\boldsymbol{\zeta}(k)])(\boldsymbol{\zeta}(k) - \mathbb{E}[\boldsymbol{\zeta}(k)])^{\mathrm{T}}\}$ which is given as

$$\boldsymbol{\Sigma}_\zeta \;=\; \varepsilon^2(1+\gamma^2)\sigma^2\begin{bmatrix} \mathbf{A}^2 & \mathbf{0}_{\mathrm{N}\times\mathrm{N}} \\ \mathbf{0}_{\mathrm{N}\times\mathrm{N}} & \mathbf{0}_{\mathrm{N}\times\mathrm{N}} \end{bmatrix}.$$

Given these definitions, we next note that

Lemma 3.4.3. *For the second-order DCTS algorithm in (3.26), the covariance matrix of the error vector* $\boldsymbol{e}(k)$ *is given as:*

$$\boldsymbol{\Sigma}_e(k) \;=\; \mathbf{P}^k\boldsymbol{e}(0)\boldsymbol{e}(0)^{\mathtt{T}}(\mathbf{P}^{\mathtt{T}})^k + \sum_{l=0}^{k-1}\mathbf{P}^l\mathbf{Q}\boldsymbol{\Sigma}_\zeta\mathbf{Q}(\mathbf{P}^{\mathtt{T}})^l, \quad (k \geq 1), \tag{3.30}$$

and the second central moment of the error vector $\boldsymbol{e}(k)$ *is given as:*

$$\kappa_e(k) \;=\; \boldsymbol{e}(0)^{\mathtt{T}}(\mathbf{P}^{\mathtt{T}})^k\mathbf{P}^k\boldsymbol{e}(0) + \mathbf{tr}\Big(\mathbf{Q}\sum_{l=0}^{k-1}(\mathbf{P}^{\mathtt{T}})^l\mathbf{P}^l\mathbf{Q}\boldsymbol{\Sigma}_\zeta\Big), \quad (k \geq 1). \tag{3.31}$$

The proof of this lemma is similar to that of Lemma 3.3.3 and thus omitted from the chapter.

(2) Asymptotic Expectation of Global Synchronization Error

Using Lemma 3.4.2, we see that the steady state of expectation of the error vector $\boldsymbol{e}(k)$ is

$$\lim_{k\to\infty}\boldsymbol{e}(k) \;=\; (1-\gamma)\varepsilon\left(\mathbf{I}_{2\mathrm{N}} - \mathbf{P}\right)^{-1}\mathbf{Q}\boldsymbol{u}_1. \tag{3.32}$$

The above equation holds because $\lim_{k\to\infty}\mathbf{P}^k = \lim_{k\to\infty}(\mathbf{H}^k - \mathbf{J}) = \mathbf{0}_{\mathrm{N}\times\mathrm{N}}$. Before we investigate the convergence property of the second-order DCTS algorithm with Gaussian delay, we give the following lemma for block matrix inversion.

Lemma 3.4.4. *Consider* $\mathrm{N}\times\mathrm{N}$ *matrices* $\mathbf{A}_1$*,* $\mathbf{A}_2$*,* $\mathbf{A}_3$ *and* $\mathbf{A}_4$*, when* $\mathbf{A}_4$ *and*

$\mathbf{C} = \mathbf{A}_1 - \mathbf{A}_2\mathbf{A}_4^{-1}\mathbf{A}_3$ *are nonsingular, then* [49]*:*

$$\begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \\ \mathbf{A}_3 & \mathbf{A}_4 \end{bmatrix}^{-1} = \begin{bmatrix} \mathbf{C}^{-1} & -\mathbf{C}^{-1}\mathbf{A}_2\mathbf{A}_4^{-1} \\ -\mathbf{A}_4^{-1}\mathbf{A}_3\mathbf{C}^{-1} & \mathbf{A}_4^{-1} + \mathbf{A}_4^{-1}\mathbf{A}_3\mathbf{C}^{-1}\mathbf{A}_2\mathbf{A}_4^{-1} \end{bmatrix}.$$

Based on this lemma, the steady state of error vector $\boldsymbol{e}(k)$ is

$$\begin{aligned} \lim_{k\to\infty} \boldsymbol{e}(k) &= (1-\gamma)\varepsilon \begin{bmatrix} \mathbf{W}_3 & \gamma\varepsilon\mathbf{W}_3\mathbf{L} \\ \tilde{\mathbf{Q}}\mathbf{W}_3 & \mathbf{I}_{\mathrm{N}} + \gamma\varepsilon\tilde{\mathbf{Q}}\mathbf{W}_3\mathbf{L} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{Q}}\boldsymbol{u} \\ \mathbf{0} \end{bmatrix} \\ &= (1-\gamma)\varepsilon \begin{bmatrix} \mathbf{W}_3\tilde{\mathbf{Q}}\boldsymbol{u} \\ \mathbf{W}_3\tilde{\mathbf{Q}}\boldsymbol{u} \end{bmatrix}, \end{aligned} \tag{3.33}$$

where $\mathbf{W}_3 = [(1-\gamma)\varepsilon\mathbf{L} + \mathbf{K}]^{-1}$. The above equation is valid because $\mathbf{K}\mathbf{W}_3 = \mathbf{K}$, which implies $\mathbf{K}\mathbf{W}_3\tilde{\mathbf{Q}} = \mathbf{0}_{\mathrm{N}\times\mathrm{N}}$, which in turn implies $\tilde{\mathbf{Q}}\mathbf{W}_3\tilde{\mathbf{Q}} = \mathbf{W}_3\tilde{\mathbf{Q}}$. Specifically, we see that the eigenvalues of $\mathbf{W}_3$ are $\lambda_1(\mathbf{W}_3) = 1$ and $\lambda_i(\mathbf{W}_3) = 1/[(1-\gamma)\varepsilon\lambda_i(\mathbf{L})],\ i = 2, \cdots, \mathrm{N}$. Additionally, the steady state of the disagreement vector $\boldsymbol{\delta}(k)$ is upper half of the vector $\lim_{k\to\infty} \boldsymbol{e}(k)$, i.e.,

$$\boldsymbol{\mu}_\infty = \lim_{k\to\infty} \boldsymbol{\delta}(k) = (1-\gamma)\varepsilon\mathbf{W}_3\tilde{\mathbf{Q}}\boldsymbol{u}. \tag{3.34}$$

For this $\boldsymbol{\mu}_\infty$, we can show that

Theorem 3.4.1. *In an undirected network with fixed, connected topology, $\boldsymbol{\mu}_\infty$ in (3.34) is a constant vector independent of the constant values of ε and γ.*

Proof. Let us denote the eigenvectors of $\mathbf{W}_3$ as $\boldsymbol{\omega}_i$. It is easy to check that the eigenvector corresponding to $\lambda_1(\mathbf{W}_3) = 1$ is $\boldsymbol{\omega}_1 = \mathbf{1}$. $\boldsymbol{\mu}_\infty$ in (3.34) can thus be rewritten as

$$\begin{aligned}\boldsymbol{\mu}_{\infty} &= (1-\gamma)\varepsilon\mathbf{1}\mathbf{1}^{\mathrm{T}}\tilde{\mathbf{Q}}\boldsymbol{u} + (1-\gamma)\varepsilon\left[\sum_{i=2}^{\mathrm{N}}\lambda_i(\mathbf{W}_3)\boldsymbol{\omega}_i\boldsymbol{\omega}_i^{\mathrm{T}}\right]\tilde{\mathbf{Q}}\boldsymbol{u} \\ &= (\mathbf{L}+\mathbf{K})^{-1}\tilde{\mathbf{Q}}\boldsymbol{u}. \qquad (3.35)\end{aligned}$$

Therefore, $\boldsymbol{\mu}_{\infty}$ does not depend on ε and γ. This completes the proof. □

Thus, for constant ε and γ, the steady state of the expectation of the disagreement vector is a constant vector regardless of ε and γ. In other words, in an undirected network with fixed topology, the expectation of global synchronization error is the same regardless of the speed of synchronization. Furthermore, we see that the steady state of disagreement vector in the first-order DCTS algorithm is exactly same as that of the second-order DCTS algorithm, then we can conclude that the second-order DCTS algorithm can improve the convergence speed without increasing the maximum asymptotic expectation of global synchronization error $\Delta t_{\max}$. Thus, we have the same definitions for the network "*average consensus achievable with tolerable synchronization error*" and "*time delay balanced network*" as the first-order DCTS algorithm.

(3) Asymptotic Mean Square Time Synchronization Error

Using Lemma 3.4.3, the steady state of the second central moment of the error vector is

$$\kappa_{e,\infty} \overset{\mathrm{def}}{=} \lim_{k\to\infty}\kappa_e(k) = \mathbf{tr}\left(\mathbf{Q}\mathbf{W}_4\mathbf{Q}\boldsymbol{\Sigma}_{\zeta}\right), \qquad (3.36)$$

where $\mathbf{W}_4 = \sum_{l=0}^{\infty}(\mathbf{P}^{\mathrm{T}})^l\mathbf{P}^l$. It is worth mentioning that $\mathbf{W}_4$ satisfies the following condition:

$$\mathbf{I}_{2\mathrm{N}} + \mathbf{P}^{\mathrm{T}}\mathbf{W}_4\mathbf{P} = \mathbf{W}_4.$$

Recall that the covariance matrix and second central moment of the disagreement vector are defined as $\mathbf{\Sigma}_\delta(k)$ and $\kappa_\delta(k)$, respectively. We see that

$$\mathbf{tr}(\mathbf{\Sigma}_e(k)) \;=\; \mathbf{tr}(\mathbf{\Sigma}_\delta(k)) + \mathbf{tr}(\mathbf{\Sigma}_\delta(k-1)).$$

Therefore, as $k \to \infty$, the steady state of second central moment of disagreement vector is

$$\kappa_{\delta,\infty} \;=\; \lim_{k\to\infty}\kappa_\delta(k) \;=\; \kappa_{e,\infty}/2 \;=\; \mathbf{tr}\left(\mathbf{Q}\mathbf{W}_4\mathbf{Q}\mathbf{\Sigma}_\zeta\right)/2.$$

Given the asymptotic mean square time synchronization error defined in (3.16), we see that in the second-order DCTS algorithm with Gaussian delay,

$$\sigma^2_{\Delta t} \;=\; \boldsymbol{u}^{\mathrm{T}}\tilde{\mathbf{Q}}\left(\mathbf{L}+\mathbf{K}\right)^{-2}\tilde{\mathbf{Q}}\boldsymbol{u} + \mathbf{tr}\left(\mathbf{Q}\mathbf{W}_4\mathbf{Q}\mathbf{\Sigma}_\zeta\right)/2. \qquad (3.37)$$

3.5 Simulation Results

The simulation parameters are described as follows: initial time phase of secondary user i is $(i-1/2)T/\mathrm{N},\ i = 1, \cdots, \mathrm{N}$, where $T = 1000\mu s$ and the standard deviation of delay variance is $\sigma = 1\mu s$. The DCTS simulation results are based on 5000 runs.

(1) First-Order DCTS with Gaussian Delay

Structured Networks

We consider the simulation for structured networks and assume $u_{cp} = 10\mu s$ and the optimal constant step size $\varepsilon_{\text{opt, FO}}$. The simulation results and asymptotic mean square time synchronization error for structured networks with 16 secondary users are shown in Fig. 3.5. The asymptotic mean square time synchronization error $\sigma^2_{\Delta t}$ is calculated from (3.17). From the plot, we see that as time index increases, mean square time synchronization error approaches the steady state value when utilizing first-order DCTS algorithm with Gaussian delay. In particular, first-order DCTS algorithm in a complete network achieves the smallest variance of synchronization error and the fastest convergence among all structured networks. This is primarily due to the high degree of connectivity in the complete network, which also results in the smallest value of $\varepsilon_{\text{opt, FO}}$. From Fig. 3.5, we see that first-order DCTS algorithm performs poorest in a path network where it has the largest value of $\sigma^2_{\Delta t}$ and the slowest convergence speed. This is not surprising since information flow from secondary user 1 to secondary user N requires N $-$ 1 hops.

Table 3.1 gives the asymptotic results for structured networks. As expected, the maximum asymptotic expectation of global time synchronization error for K_{N}, R_{N} and H_{N} is 0 as discussed earlier. It is worth mentioning that first-order DCTS algorithm in hypercube network H_{N} has a relatively small value of $\sigma^2_{\Delta t}$ because of its symmetric and well-connected structure. Furthermore, we see that the first-order DCTS algorithm in S_{N} performs slightly worse than H_{N}. In fact, first-order DCTS algorithm for a star network can be viewed as a type of centralized

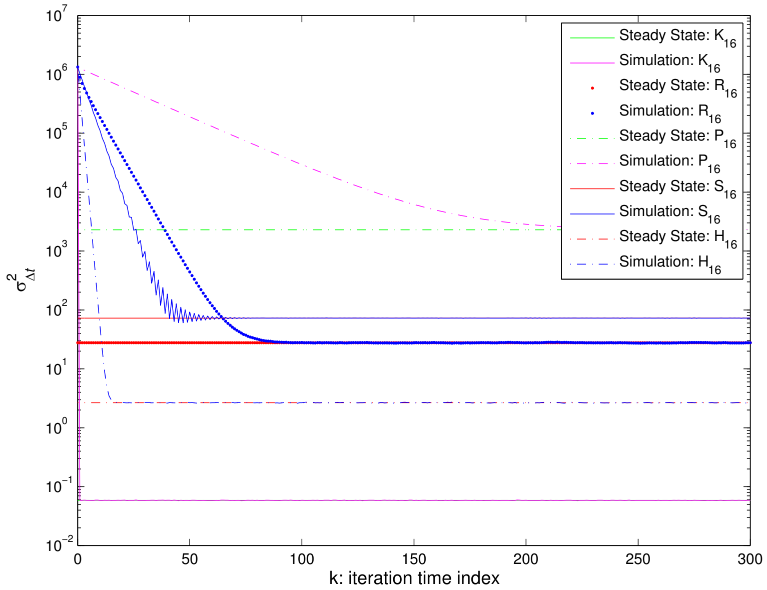


Figure 3.5: $\sigma^2_{\Delta t}$ as a function of iteration time index for the first-order DCTS algorithm in structured networks (K_{16}, R_{16}, P_{16}, S_{16}, H_{16}) with Gaussian delay.

time synchronization algorithm, in which a root secondary user determines and propagates the average of local time information of all other secondary users in the networks.

Table 3.1: Asymptotic Results for First-order DCTS Algorithm in Structured Networks with Gaussian Delay

	K_{16}	R_{16}	P_{16}	S_{16}	H_{16}
$\Delta t_{\max}(\mu s)$	0	0	35	8.75	0
$\sigma^2_{\Delta t}$	0.0586	27.7429	2295.2	72.7148	2.6667

In Fig. 3.6, we show the asymptotic value of $\sigma^2_{\Delta t}$ as a function of the number of secondary users in the various structured networks. We see that when using the optimal constant step size, the asymptotic mean square time synchronization error for a star network is nearly constant as the number of secondary users increases; we observe a similar trend in Section 3.3.3. Furthermore, as stated in Section 3.3.3, for a complete network, $\sigma^2_{\Delta t}$ decreases as number of secondary users increases. On the contrary, $\sigma^2_{\Delta t}$ is an increasing function of the number of secondary users for both path and ring networks.

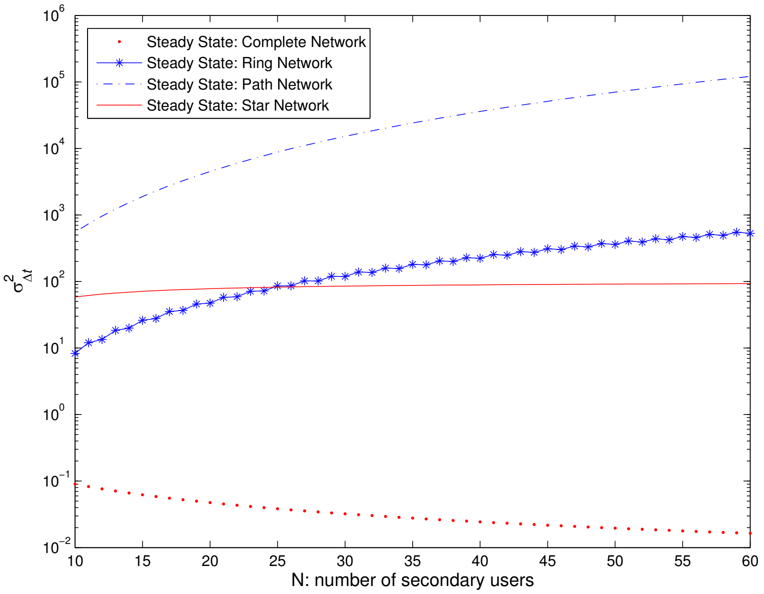


Figure 3.6: $\sigma^2_{\Delta t}$ as a function of the number of secondary users for the first-order DCTS algorithm in structured networks with Gaussian delay.

Random Networks

Fig. 3.7 shows the simulation results when the first-order DCTS algorithm is implemented in a random network of Fig. 2.4(b) assuming Gaussian delay between secondary users. In the simulation, we choose $\varepsilon_{\text{opt, FO}}$ and assume the average distance between two secondary users is $0.5km$. From the plot, we see that asymptotically, there exists global synchronization error between some pairs of secondary users, and $\Delta t_{\max} = 26.4130\mu s$ for this random network. If we specify a threshold

Δt_{Th} to be greater than or equal to this $\Delta t_{\max}$, we call this network as "*average consensus achievable with tolerable synchronization error*" as described in Definition 3.3.1.

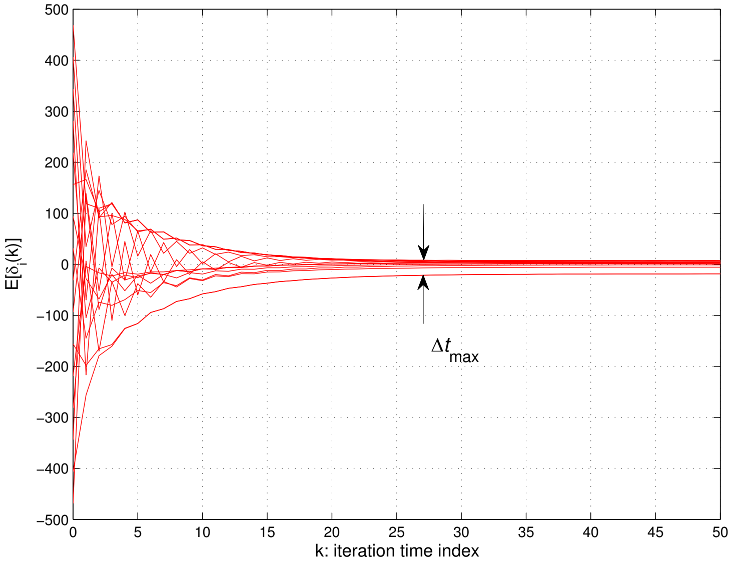


Figure 3.7: Evolution of the average disagreement of the first-order DCTS algorithm in random network (Fig. 2.4(b)) with Gaussian delay.

(2) Second-Order DCTS Algorithm with Gaussian Delay

Structured Networks

In our simulations of the second-order DCTS algorithm with Gaussian delay, we assume $u_{cp} = 10\mu s$ and the optimal values of $\varepsilon_{\text{opt, SO}}$ and $\gamma_{\text{opt, SO}}$ as discussed in Chapter 2. The simulation results and the asymptotic mean square time synchronization errors for the K_{16}, R_{16}, P_{16}, S_{16} and H_{16} networks are shown in Fig. 3.8. For each network topology, the asymptotic mean square time synchronization error $\sigma^2_{\Delta t}$ is calculated from (3.37). We see that as time index increases, mean square time synchronization error approaches the steady state value when utilizing second-order DCTS algorithm with Gaussian delay. Similar to the first-order DCTS algorithm, the second-order DCTS algorithm performs poorest in a path network where it has the largest value of $\sigma^2_{\Delta t}$ and the slowest convergence speed. Interestingly, the second-order DCTS algorithm behaves exactly same as the first-order DCTS algorithm in a complete network as shown in Fig. 3.5. This is due to the fact that for a complete network, $\lambda_2(\mathbf{L}) = \lambda_{\mathrm{N}}(\mathbf{L}) = \mathrm{N}$, and when utilizing the optimal values of $\varepsilon_{\mathrm{opt,\ SO}}$ and $\gamma_{\mathrm{opt,\ SO}}$, we have $\gamma_{\mathrm{opt,\ SO}} = 0$, which implies that the second-order DCTS algorithm reduces to the first-order DCTS algorithm.

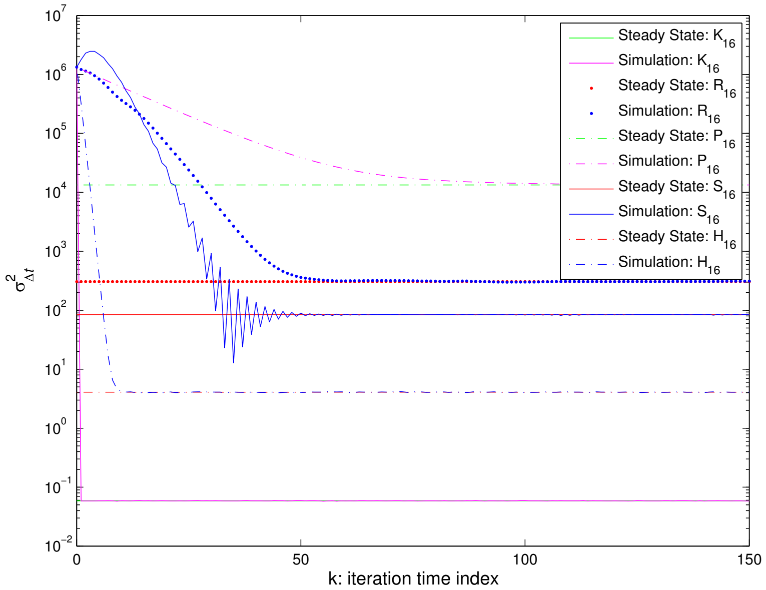


Figure 3.8: $\sigma^2_{\Delta t}$ as a function of iteration time index for the second-order DCTS algorithm in structured networks (K_{16}, R_{16}, P_{16}, S_{16}, H_{16}) with Gaussian delay.

Table 3.2 summarizes the asymptotic results of the second-order DCTS algorithm for structured networks. As expected, the maximum asymptotic expectation of global time synchronization error for K_{N}, R_{N} and H_{N} is 0 since K_{N}, R_{N} and H_{N} are *time delay balanced networks*. Furthermore, the second-order DCTS algorithm in P_{N} has the largest $\Delta t_{\max}$ because of its highly unbalanced time delay structure. Similar to the first-order DCTS algorithm, the second-order DCTS algorithm in hypercube network H_{N} has relatively small values of $\sigma^2_{\Delta t}$.

Table 3.2: Asymptotic Results for Second-Order DCTS Algorithm in Structured Networks with Gaussian Delay

	K_{16}	R_{16}	P_{16}	S_{16}	H_{16}
$\Delta t_{\max}(\mu s)$	0	0	35	8.75	0
$\sigma^2_{\Delta t}$	0.0586	304075	13329	84.2996	4.0850

In Fig. 3.9, we show the asymptotic value of $\sigma^2_{\Delta t}$ as a function of the number of secondary users in these structured networks. From the plot, we see that when using the optimal $\varepsilon_{\text{opt, SO}}$ and $\gamma_{\text{opt, SO}}$, the asymptotic mean square time synchronization error for a star network is nearly constant as the number of secondary users increases. However, $\sigma^2_{\Delta t}$ is an increasing function of the number of secondary users for both path and ring networks.

Random Networks

Fig. 3.10 shows the simulation results when the second-order DCTS algorithm is implemented in a random network of Fig. 2.4(b) assuming Gaussian delay between secondary users. We see here that an asymptotic global synchronization error persists between some pairs of secondary users, i.e., $\Delta t_{\max} = 26.4130\mu s$ for this random network. If we specify a threshold Δt_{Th} to be greater than or equal to

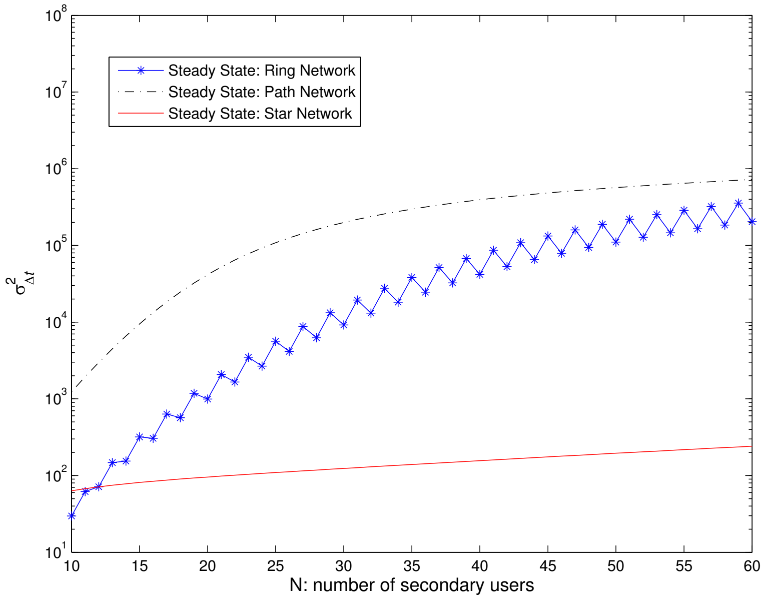


Figure 3.9: $\sigma^2_{\Delta t}$ as a function of the number of secondary users for the second-order DCTS algorithm in structured networks with Gaussian delay.

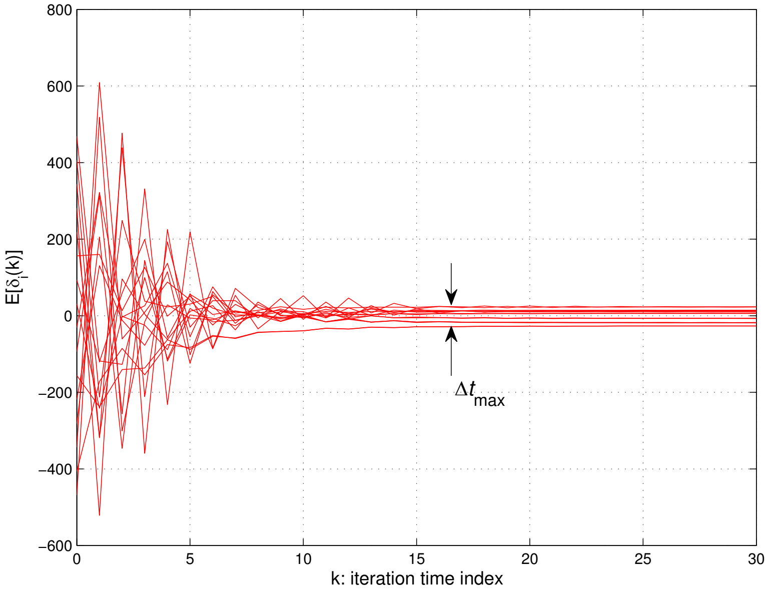


Figure 3.10: Evolution of the average disagreement of the second-order DCTS algorithm in random network (Fig. 2.4(b)) with Gaussian delay.

this $\Delta t_{\max}$, we call this network as "*average consensus achievable with tolerable synchronization error*" as described in Definition 3.3.1. It should be noted that the first-order DCTS algorithm has the same asymptotic global synchronization error as the second-order DCTS algorithm.

3.6 Chapter Summary

In this chapter, we have presented theoretical results on the convergence of the first- and second-order DCTS algorithm for cognitive radio networks with general Gaussian delay between secondary users. Specifically, we have computed the asymptotic expectation and mean square of the global synchronization error of the DCTS algorithm. The results lead to the definition of a *time delay balanced network* in which average timing consensus between secondary users can be achieved despite random delays. Furthermore, several structured network architectures have been studied as examples and their associated simulation results have been used to validate analytical findings.

Chapter 4

Cooperative Spectrum Sensing in Cost Constrained Cognitive Radio Networks: Parallel Access Channels

4.1 Introduction

In this chapter, we study the cooperative spectrum sensing in cost constrained cognitive radio network over PAC channels. As spectrum sensing proposes significant overhead in the performance of cognitive radios, it must be conducted while balancing its gains against its costs. Low-energy overhead cooperative spectrum sensing was studied in [40]. Optimally allocated powers were computed without taking into account the underlying *system level* cost of sensing. Our work on energy-constrained spectrum sensing is motivated by [59], in which detection problems were formulated to account for constraints on expected cost due to transmission and measurement. For cost constrained cognitive radio networks, we build on these formulations here to design optimal strategy for cooperative spectrum sensing.

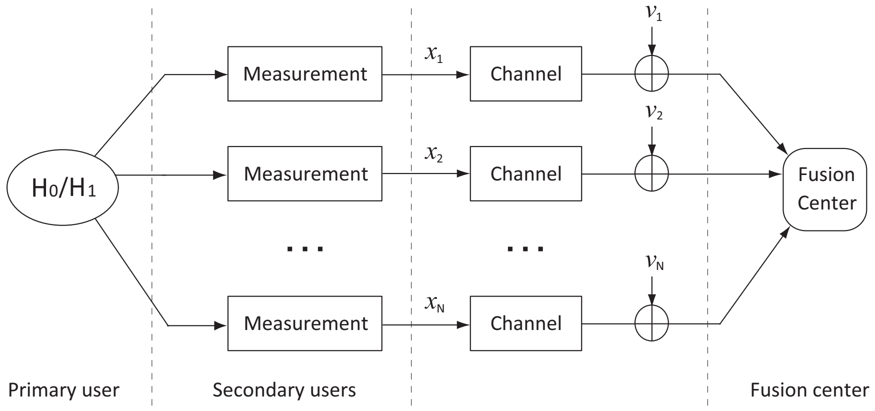


Figure 4.1: Cooperative spectrum sensing in cognitive radio networks over PAC.

As shown in Fig. 4.1, secondary users forward local energy detection statistics to a secondary base station using amplify-and-forward (AF) over PAC. Based on this model, we account for two major factors that contribute to the system level energy cost of sensing: Local processing cost due to sample collection and local energy calculation and transmission due to forwarding local statistic to the fusion center. We present two optimization problems to maximize the global detection probability by choosing the appropriate number of energy samples that must be collected at each secondary user and the appropriate amplifier gain that each secondary user must use to forward its statistics to the fusion center. When jointly designing the number of samples and amplifier gains, we demonstrate that only *one* secondary user needs to be actively processing and transmitting local statistics to the fusion center, i.e., only *one* secondary user must conduct spectrum sensing to achieve optimal performance. When either the amplifier gains or the number of samples are fixed, we derive closed-form expressions for optimal solutions and propose a generalized water-filling approach for cost constrained cooperative spectrum sensing.

The remainder of this chapter is outlined as follows: Section 4.2 describes the system model. Sections 4.3 and 4.4 collectively present our results for energy-constrained spectrum sensing: Section 4.3 discusses the optimization for maximization of global detection probability while Section 4.4 provides the optimization for minimization of system level cost. Simulation results are presented in Section 4.5.

4.2 System Model

(1) Communication Model

We consider a network model in Fig. 4.1, where secondary user conducts local spectrum sensing and transmits its local energy statistic to the fusion center using AF on PAC. The received signal at the fusion center is shown in Fig. 4.1, i.e.,

$$y_i \;=\; g_i h_i x_i + v_i, \tag{4.1}$$

where x_i is the energy of received signal at the secondary user i; g_i is the amplifier gain for the secondary user i; h_i is channel gain between secondary user i and and the fusion center and v_i is independent and identically distributed (i.i.d.) Gaussian noise, i.e., $v_i \sim \mathcal{N}(0, \sigma_v^2)$ and is independent of x_i. We assume that h_i is known at the fusion center (e.g., via channel estimation) and remains constant during the sensing period. We can then rewrite (4.1) in a matrix form as

$$\boldsymbol{y} \;=\; \boldsymbol{\Omega}\boldsymbol{x} + \boldsymbol{v}, \tag{4.2}$$

where $\boldsymbol{x} = [x_1, x_2, \cdots, x_{\mathrm{N}}]^{\mathtt{T}}$, $\boldsymbol{y} = [y_1, y_2, \cdots, y_{\mathrm{N}}]^{\mathtt{T}}$, $\boldsymbol{v} = [v_1, v_2, \cdots, v_{\mathrm{N}}]^{\mathtt{T}}$ and $\boldsymbol{\Omega} = \mathrm{diag}\{g_1 h_1, g_2 h_2, \cdots, g_{\mathrm{N}} h_{\mathrm{N}}\}$.

(2) Local Energy Statistic

For secondary user i, $(1 \le i \le \mathrm{N})$, the hypothesis test for x_i is

$$\begin{cases} \mathcal{H}_0 : & x_i \;=\; (1/\kappa_i)\sum_{k=1}^{\kappa_i} |n_i(k)|^2 \\ \mathcal{H}_1 : & x_i \;=\; (1/\kappa_i)\sum_{k=1}^{\kappa_i} |\tilde{h}_i s(k) + n_i(k)|^2, \end{cases} \tag{4.3}$$

where κ_i is the number of samples, $s(k)$ is the transmitted signal from the primary user and $n_i(k)$ is the noise received by secondary user i. We assume $s(k)$ is complex PSK modulated and i.i.d. with mean zero and variance σ_s^2; $\tilde{h}_i$ is the channel gain between the primary user and secondary user i and is assumed to be constant during the cooperative spectrum sensing period; and $n_i(k)$ is i.i.d. circularly symmetric complex Gaussian random variable with mean zero and variance σ_n^2 and is independent of $s(k)$. We define the local received SNR at the secondary user i as $\gamma_i = \sigma_s^2 |\tilde{h}_i|^2 / \sigma_n^2$.

When κ_i is large, x_i can be approximated as Gaussian random variable [25,60], i.e.,

$$\begin{cases} \mathcal{H}_0: & x_i \;\sim\; \mathcal{N}(\sigma_n^2,\ \sigma_n^4/\kappa_i) \\ \mathcal{H}_1: & x_i \;\sim\; \mathcal{N}((1+\gamma_i)\sigma_n^2,\ (1+2\gamma_i)\sigma_n^4/\kappa_i). \end{cases} \tag{4.4}$$

In this chapter, we assume the local received SNR γ_i is known at the secondary user i. For instance, in IEEE 802.22, this value could be obtained through estimation of pilot signals periodically transmitted from TV stations [61].

Given this system model, we see that

$$\xi_i \overset{\text{def}}{=} \mathbb{E}\{x_i^2\} \;=\; [1 + 1/\kappa_i + \pi_1 (\gamma_i + 2(1+1/\kappa_i))\gamma_i]\,\sigma_n^4,$$

where $\pi_0 = \mathrm{P}(\mathcal{H}_0)$ and $\pi_1 = \mathrm{P}(\mathcal{H}_1)$ are the probabilities that spectrum is idle and occupied, respectively. In cognitive radio networks, the received primary user power measured by the secondary user is expected to be very small [62], i.e., $\gamma_i \ll 1$. Additionally, the number of samples is expected to be more than a few, i.e., $\kappa_i \gg 1$. Then, we can approximate the transmitted power for the secondary user i as $\mathcal{P}_i = \xi_i g_i^2 \simeq g_i^2(1+2\pi_1\gamma_i)\sigma_n^4$.

(3) Neyman-Pearson (NP) Detection Rule

Under hypothesis $\mathcal{H}_0$ and $\mathcal{H}_1$, the received signal $\boldsymbol{y}$ has a Gaussian distribution, i.e.,

$$\begin{cases} \mathcal{H}_0: & \boldsymbol{y} \;\sim\; \mathcal{N}(\boldsymbol{\Omega}\mathbf{1}\sigma_n^2,\ \boldsymbol{\Sigma}_0) \\ \mathcal{H}_1: & \boldsymbol{y} \;\sim\; \mathcal{N}(\boldsymbol{\Omega}(\mathbf{1}+\boldsymbol{\gamma})\sigma_n^2,\ \boldsymbol{\Sigma}_1), \end{cases} \tag{4.5}$$

where $\boldsymbol{\Sigma}_0 = \sigma_n^4\boldsymbol{\Omega}\mathbf{U}^{-1}\boldsymbol{\Omega}^\dagger + \sigma_v^2\mathbf{I}_\mathrm{N}$, here, $\mathbf{U} = \mathrm{diag}\{\kappa_1, \kappa_2, \cdots, \kappa_\mathrm{N}\}$; $\boldsymbol{\Sigma}_1 = \sigma_n^4\boldsymbol{\Omega}\mathbf{U}^{-1}(\mathbf{I}_\mathrm{N} + 2\boldsymbol{\Gamma})\boldsymbol{\Omega}^\dagger + \sigma_v^2\mathbf{I}_\mathrm{N}$, here $\boldsymbol{\Gamma} = \mathrm{diag}\{\gamma_1, \gamma_2, \cdots, \gamma_\mathrm{N}\}$; and $\boldsymbol{\gamma} = [\gamma_1, \gamma_2, \cdots, \gamma_\mathrm{N}]^\mathtt{T}$.

Since $\gamma_i \ll 1$ and $\kappa_i \gg 1$, then, $\gamma_i/\kappa_i \approx 0$ and we have $\boldsymbol{\Sigma}_0 \approx \boldsymbol{\Sigma}_1$. Thus, the optimal LRT can be approximated as

$$\mathcal{T}(\boldsymbol{y}) \;=\; (\boldsymbol{\Omega\gamma})^\dagger \boldsymbol{\Sigma}_0^{-1}\boldsymbol{y} \underset{\mathcal{H}_0}{\overset{\mathcal{H}_1}{\gtrless}} \tau_g, \tag{4.6}$$

where τ_g is the global decision threshold. Furthermore, we note that

$$\begin{cases} \mathbb{E}\{\mathcal{T}(\boldsymbol{y})|\mathcal{H}_0\} \;=\; (\boldsymbol{\Omega\gamma})^\dagger \boldsymbol{\Sigma}_0^{-1}\boldsymbol{\Omega}\mathbf{1}\sigma_n^2 \\ \mathbb{E}\{\mathcal{T}(\boldsymbol{y})|\mathcal{H}_1\} \;=\; (\boldsymbol{\Omega\gamma})^\dagger \boldsymbol{\Sigma}_0^{-1}\boldsymbol{\Omega}(\mathbf{1}+\boldsymbol{\gamma})\sigma_n^2, \end{cases}$$

and

$$\mathbb{V}\text{ar}\{\mathcal{T}(\boldsymbol{y})|\mathcal{H}_0\} \;=\; \mathbb{V}\text{ar}\{\mathcal{T}(\boldsymbol{y})|\mathcal{H}_1\} \;=\; (\boldsymbol{\Omega\gamma})^\dagger \boldsymbol{\Sigma}_0^{-1}\boldsymbol{\Omega\gamma}.$$

After some steps, for NP detection with false alarm probability $\mathrm{P}_f = \alpha$, the global decision threshold can be obtained as

$$\tau_g \;=\; (\boldsymbol{\Omega\gamma})^\dagger \boldsymbol{\Sigma}_0^{-1}\boldsymbol{\Omega}\mathbf{1}\sigma_n^2 + \mathcal{Q}^{-1}(\alpha)\big[(\boldsymbol{\Omega\gamma})^\dagger \boldsymbol{\Sigma}_0^{-1}\boldsymbol{\Omega\gamma}\big]^{1/2},$$

where $\mathcal{Q}(x)$ is the complementary distribution function of the standard Gaussian, i.e., $\mathcal{Q}(x) = \frac{1}{\sqrt{2\pi}}\int_x^\infty \exp(-t^2/2)\mathrm{d}t$. The global detection probability is then given as

$$\begin{aligned} \mathrm{P}_d \;&=\; \mathcal{Q}\Big(\mathcal{Q}^{-1}(\alpha) - \sigma_n^2\big[(\boldsymbol{\Omega\gamma})^\dagger \boldsymbol{\Sigma}_0^{-1}\boldsymbol{\Omega\gamma}\big]^{1/2}\Big) \\ &=\; \mathcal{Q}\left(\mathcal{Q}^{-1}(\alpha) - \left(\sum_{i=1}^{\mathrm{N}} \frac{g_i^2\kappa_i\gamma_i^2|h_i|^2}{g_i^2|h_i|^2+\kappa_i\tilde{\sigma}_v^2}\right)^{1/2}\right), \end{aligned} \tag{4.7}$$

where $\tilde{\sigma}_v^2 = \sigma_v^2/\sigma_n^4$. It is also easy to see that the asymptotic detection probability expressions when the number of samples or amplifier gains approach infinity are given by

$$\mathrm{P}_d(\kappa_\infty) \overset{\text{def}}{=} \lim_{\kappa_i \to \infty} \mathrm{P}_d = \mathcal{Q}\left(\mathcal{Q}^{-1}(\alpha) - \frac{1}{\tilde{\sigma}_v}\left(\sum_{i=1}^{\mathrm{N}} g_i^2 \gamma_i^2 |h_i|^2\right)^{1/2}\right), \qquad (4.8)$$

and

$$\mathrm{P}_d(g_\infty) \overset{\text{def}}{=} \lim_{g_i \to \infty} \mathrm{P}_d = \mathcal{Q}\left(\mathcal{Q}^{-1}(\alpha) - \left(\sum_{i=1}^{\mathrm{N}} \kappa_i \gamma_i^2\right)^{1/2}\right), \qquad (4.9)$$

respectively.

(4) System Level Cost for Cooperative Spectrum Sensing

In this chapter, we consider system level cost for cooperative spectrum sensing in cognitive radio networks. The system level cost consists of three parts: local processing cost, transmission cost, reporting and broadcasting cost.

- Local processing cost includes the receiver RF scanning and local energy calculation. For simplicity, we assume that the local processing cost $\mathcal{C}_{pi}(\cdot)$ for secondary user i is a linear function of the number of samples, i.e.,

  $$\mathcal{C}_{pi}(\kappa_i) = c_0 \kappa_i,$$

  where c_0 is the local processing cost per sample.

- Transmission cost is the transmit power required from a secondary user to transmit the local calculated energy to the fusion center. Here, we assume that this cost for secondary user i is given as

$$\mathcal{C}_{ti}(g_i) = \mathcal{P}_i = \xi_i g_i^2.$$

- For optimal system design, the fusion center needs to know the local received SNR for each secondary user. In practice, this means that secondary users will report their local received SNRs to the fusion center. The fusion center then determines the resource allocated to each secondary user, and broadcasts this to all secondary users. In this chapter, we assume that total reporting and broadcasting cost $\mathcal{C}_{rb}$ is fixed, and thus do not consider it in the optimization problem.

The system level cost during the cooperative spectrum sensing is given as

$$\begin{aligned} \mathcal{C}(\boldsymbol{\kappa}, \boldsymbol{g}) &= \sum_{i=1}^{\mathrm{N}} \mathcal{C}_{pi}(\kappa_i) + \sum_{i=1}^{\mathrm{N}} \mathcal{C}_{ti}(g_i) \\ &= \sum_{i=1}^{\mathrm{N}} \left(c_0 \kappa_i + \xi_i g_i^2 \right), \end{aligned}$$

where $\boldsymbol{\kappa} = [\kappa_1, \kappa_2, \cdots, \kappa_{\mathrm{N}}]^{\mathrm{T}}$ and $\boldsymbol{g} = [g_1, g_2, \cdots, g_{\mathrm{N}}]^{\mathrm{T}}$.

4.3 Optimization: Maximization of Detection Probability

In this section, we aim to maximize the detection probability for the system model in Fig. 4.1 subject to a system level cost constraint of sensing. Specifically, we determine the appropriate number of samples and amplifier gains for each secondary user and consider the following two scenarios for this optimization problem:

1. **Scenario A:** First, we consider the system level cost constraint. Hence, the optimization problem is formulated as:

$$\begin{aligned} \max_{\boldsymbol{\kappa},\boldsymbol{g}} \quad & \mathrm{P}_d(\boldsymbol{\kappa},\boldsymbol{g}) \\ \text{s.t.} \quad & \mathcal{C}(\boldsymbol{\kappa},\boldsymbol{g}) \ \leq \ \bar{\mathcal{C}}, \\ & \boldsymbol{\kappa} \in \mathcal{Z}_+^{\mathrm{N}},\ \boldsymbol{g} \in \mathcal{R}_+^{\mathrm{N}}, \end{aligned} \tag{4.10}$$

   where $\bar{\mathcal{C}}$ is the system level cost constraint. Here we denote the optimal solution of (4.10) as $\left(\kappa_{p,i}^{(\mathrm{opt},1)}, g_{p,i}^{(\mathrm{opt},1)}\right)$ and the maximum detection probability as $\mathrm{P}_d^{(\mathrm{opt},1)}$.

2. **Scenario B:** In some applications, local sample collection for each secondary user may be scheduled in a fixed time slot. In other words, the number of samples is upper bounded by a maximum value $\kappa_{\max}$. Furthermore, the transmission power for each secondary user may be required to be below a predefined power limit $\mathcal{P}_{\max}$. By incorporating these additional individual constraints imposed on each secondary user, we can model the optimization problem as

$$\begin{aligned} \max_{\boldsymbol{\kappa},\boldsymbol{g}} \quad & \mathrm{P}_d(\boldsymbol{\kappa},\boldsymbol{g}) \\ \text{s.t.} \quad & \mathcal{C}(\boldsymbol{\kappa},\boldsymbol{g}) \leq \bar{\mathcal{C}},\ \boldsymbol{\kappa} \in \mathcal{Z}_+^{\mathrm{N}},\ \boldsymbol{g} \in \mathcal{R}_+^{\mathrm{N}}, \\ & \boldsymbol{\kappa} \preceq \kappa_{\max}\mathbf{1},\ \xi_i g_i^2 \leq \mathcal{P}_{\max}. \end{aligned} \tag{4.11}$$

To better understand the optimal resource allocation for cooperative spectrum sensing, we consider the following two cases in Scenarios A and B: joint optimization of $\boldsymbol{\kappa}$ and $\boldsymbol{g}$; and optimization of either $\boldsymbol{\kappa}$ or $\boldsymbol{g}$.

4.3.1 Case I: Joint Optimization of Number of Samples and Amplifier Gains

(1) Scenario A

In this case, we consider the optimization in (4.10) over both $\boldsymbol{\kappa}$ and $\boldsymbol{g}$. We note that (4.10) is a mixed integer nonlinear optimization problem (MINLP). In general, there is no polynomial-time algorithm for solving general MINLPs [63]. A potentially clearer insight into the solutions can be obtained by considering a convex relaxation for this optimization problem, where we simply relaxed the integer constraint of the number of samples:

$$\begin{aligned} \max_{\boldsymbol{\kappa},\boldsymbol{g}} \quad & \mathrm{P}_d(\boldsymbol{\kappa},\boldsymbol{g}) \\ \text{s.t.} \quad & \mathcal{C}(\boldsymbol{\kappa},\boldsymbol{g}) \ \leq \ \bar{\mathcal{C}}, \\ & \boldsymbol{\kappa} \in \mathcal{R}_+^{\mathrm{N}}, \ \boldsymbol{g} \in \mathcal{R}_+^{\mathrm{N}}. \end{aligned} \tag{4.12}$$

Let us define $z_i = g_i^2$, $p_i = \tilde{\sigma}_v^2/(\gamma_i^2|h_i|^2)$ and $q_i = 1/\gamma_i^2$. To simplify our analysis, when $\kappa_i = z_i = 0$, we assume $\kappa_i z_i/(p_i\kappa_i + q_i z_i) = 0$. In practice, this assumption can be alleviated by adding a sufficiently small constant in the denominator.

Plugging (4.7) into (4.12), the optimization problem becomes

$$\begin{aligned} \max_{\boldsymbol{\kappa},\boldsymbol{z}} \quad & \sum_{i=1}^{\mathrm{N}} \frac{\kappa_i z_i}{p_i \kappa_i + q_i z_i} \\ \text{s.t.} \quad & c_0 \mathbf{1}^{\mathrm{T}} \boldsymbol{\kappa} + \boldsymbol{\xi}^{\mathrm{T}} \boldsymbol{z} \ \leq \ \bar{\mathcal{C}}, \\ & \boldsymbol{\kappa} \ \succeq \ \mathbf{0}, \ \boldsymbol{z} \ \succeq \ \mathbf{0}. \end{aligned} \tag{4.13}$$

where $\boldsymbol{z} = [z_1, z_2, \cdots, z_{\mathrm{N}}]^{\mathrm{T}}$ and $\boldsymbol{\xi} = [\xi_1, \xi_2, \cdots, \xi_{\mathrm{N}}]^{\mathrm{T}}$. Then, we note that

Lemma 4.3.1. *(4.13) is a convex optimization problem.*

Proof. Let us define

$$\mathcal{F}_i(\kappa_i, z_i) \ = \ \frac{\kappa_i z_i}{p_i \kappa_i + q_i z_i}.$$

After some manipulations, we see that the Hessian of $\mathcal{F}_i(\kappa_i, z_i)$ is given as

$$\nabla^2 \mathcal{F}_i(\kappa_i, z_i) \ = \ -\frac{2 p_i q_i}{(p_i \kappa_i + q_i z_i)^3} \begin{bmatrix} z_i \\ \kappa_i \end{bmatrix} \begin{bmatrix} z_i \\ \kappa_i \end{bmatrix}^{\mathrm{T}} \ \preceq \ \mathbf{0}.$$

Thus, $\mathcal{F}_i(\kappa_i, z_i)$ is a concave function, which indicates that the objective function in (4.13) is also concave. This completes the proof. □

Since (4.13) is a convex problem, it can be solved efficiently using interior-point methods or other iterative methods [64]. This will be a recurring theme in the optimization problems we consider in the sequel. In the numerical results, we shall see that the approximation as detailed below results in near optimal performance without the curse of complexity. Given this convex optimization problem, first we introduce the following lemma.

Lemma 4.3.2. *Optimal solution of* $(\boldsymbol{\kappa}, \boldsymbol{z})$ *in (4.13) should satisfy either 1)* $\kappa_i > 0$ *and* $z_i > 0$*, or 2)* $\kappa_i = 0$ *and* $z_i = 0$ *for secondary user* i*.*

Proof. Please see the Appendix C.1. □

This lemma is not surprising because when one secondary user does not collect the energy samples, it will not have anything to transmit to the fusion center. Similarly, when one secondary user decides not to transmit the data to the fusion center, it is reasonable to expect that this secondary user should remain inactive and not collect local energy samples. Using Lemma 4.3.2, the optimal solution of $(\boldsymbol{\kappa}, \boldsymbol{g})$ can be found as stated in the following theorem.

Theorem 4.3.1. *Consider the optimization problem in (4.13), let us define*

$$\rho_i = \frac{\gamma_i^2 |h_i|^2}{(\tilde{\sigma}_v \sqrt{\xi_i} + |h_i| \sqrt{c_0}\,)^2} \tag{4.14}$$

and assume $\rho_1 \geq \rho_2 \geq \cdots \geq \rho_{\mathrm{N}}$*. Then, the optimal solution of* $(\boldsymbol{\kappa}, \boldsymbol{g})$ *is*

$$\begin{aligned} \kappa_{p,i}^{(\mathrm{opt},2)} &= \begin{cases} \frac{|h_i| \bar{\mathcal{C}}}{\tilde{\sigma}_v \sqrt{\xi_i c_0} + |h_i| c_0}, & i = 1 \\ 0, & i > 1, \end{cases} \\ g_{p,i}^{(\mathrm{opt},2)} &= \begin{cases} \left(\frac{\tilde{\sigma}_v \bar{\mathcal{C}}}{\tilde{\sigma}_v \xi_i + |h_i| \sqrt{\xi_i c_0}} \right)^{1/2}, & i = 1 \\ 0, & i > 1. \end{cases} \end{aligned} \tag{4.15}$$

Proof. Please see the Appendix C.2. □

Given the optimal solution of $(\boldsymbol{\kappa}, \boldsymbol{g})$, the optimal detection probability in (4.13) is

$$\mathrm{P}_d^{(\mathrm{opt},2)} \;=\; \mathcal{Q}\left(\mathcal{Q}^{-1}(\alpha) - \sqrt{\bar{\mathcal{C}}}\max\left\{\frac{\gamma_i|h_i|}{\tilde{\sigma}_v\sqrt{\xi_i}+|h_i|\sqrt{c_0}}\right\}\right).$$

Since (4.12) is the relaxation of the MINLP (4.10), we see that $\mathrm{P}_d^{(\mathrm{opt},1)} \leq \mathrm{P}_d^{(\mathrm{opt},2)}$ [63]. In practice, we may consider a floor operation for the number of samples as a suboptimal solution for (4.10), i.e.,

$$\kappa_{p,i}^{(\mathrm{sub})} = \left\lfloor \kappa_{p,i}^{(\mathrm{opt},2)} \right\rfloor \quad \text{and} \quad g_{p,i}^{(\mathrm{sub})} = g_{p,i}^{(\mathrm{opt},2)},\ \forall i. \tag{4.16}$$

Let us denote the resulting global detection probability as $\mathrm{P}_d^{(\mathrm{sub})}$. Then we see that $\mathrm{P}_d^{(\mathrm{opt},2)} \geq \mathrm{P}_d^{(\mathrm{opt},1)} \geq \mathrm{P}_d^{(\mathrm{sub})}$. Furthermore, when $\kappa_{p,1}^{(\mathrm{opt},2)}$ is large, based on the first-order Taylor series, we have

$$\begin{aligned}
\mathrm{P}_d^{(\mathrm{opt},2)} - \mathrm{P}_d^{(\mathrm{sub})} \;&=\; \mathrm{P}_d(\boldsymbol{\kappa},\boldsymbol{g}) - \mathrm{P}_d(\boldsymbol{\kappa}-\Delta\boldsymbol{\kappa},\boldsymbol{g}) \\
&\approx\; \frac{\Delta\kappa_1\delta_0\delta_1}{2\sqrt{2\pi}}\exp\left(-\frac{1}{2}\left(\mathcal{Q}^{-1}(\alpha)-\delta_0\right)^2\right)(\kappa_1+\delta_1)^{-2} \\
&\to\; 0^+,
\end{aligned}$$

where $\delta_0 = g_1\gamma_1|h_1|/\tilde{\sigma}_v$ and $\delta_1 = g_1^2|h_1|^2/\tilde{\sigma}_v$. With small value of $\Delta\kappa_1$ (normally $\Delta\kappa_1 < 1$), it is interesting to note that our rounding algorithm is near optimal with large system level cost constraint. When $\bar{\mathcal{C}}$ is relatively small, as we will show in our simulations, our proposed suboptimal algorithm can also provide a good approximation to the optimal solution.

Based on (4.16), when we jointly design the number of samples and amplifier gains subject to the system level cost constraint, only *one* secondary user needs to be active in the cognitive radio network, i.e., collecting local energy samples and transmitting the energy statistic to the fusion center. It is interesting to note that this strategy is similar to multiuser diversity where the base station selects the user with the highest channel to achieve maximum sum rate capacity [65]. In this case, the fusion center will select the secondary user with the largest ρ_i to perform local spectrum sensing and data forwarding. This will significantly reduce the bandwidth cost for data forwarding.

Remark: We note that the result in (4.16) can be implemented in a distributed fashion. The idea is based on opportunistic carrier sensing [10] or opportunistic relaying [66] in which a backoff timer is set to be a decreasing function of channel state information. In particular, at the beginning of each sensing time slot, the fusion center broadcasts a beacon signal to synchronize all secondary users in the cognitive radio network. After estimating the channel gain[1] $|h_i|$, the secondary user calculates the control parameter ρ_i based on its local received SNR γ_i and then maps ρ_i to a backoff timer $f(\rho_i)$ (equal to c/ρ_i in [66], where c is a constant). Under a collision free situation, the secondary user with largest ρ_i will expire first and perform local energy calculation and data forwarding during this time slot[2]. Note that in this case, fusion center does not need to broadcast the optimal design parameter for each secondary user and this will reduce the cooperative sensing cost for broadcasting and reporting.

[1] We assume reciprocity of the uplink and downlink channels between the fusion center and secondary users [67].

[2] Detailed analysis on how to reduce the collision probability for this scheme can be found in [10].

(2) Scenario B

We examine the optimization (4.11) over both $\boldsymbol{\kappa}$ and $\boldsymbol{g}$. Similar to Scenario A, we first consider the relaxation to the original MINLP in (4.11), i.e.,

$$\begin{aligned} \max_{\boldsymbol{\kappa},\boldsymbol{z}} \quad & \sum_{i=1}^{\mathrm{N}} \frac{\kappa_i z_i}{p_i \kappa_i + q_i z_i} \\ \text{s.t.} \quad & c_0 \mathbf{1}^{\mathrm{T}} \boldsymbol{\kappa} + \boldsymbol{\xi}^{\mathrm{T}} \boldsymbol{z} \ \le \ \bar{\mathcal{C}}, \\ & \mathbf{0} \ \preceq \ \boldsymbol{\kappa} \ \preceq \ \kappa_{\max} \mathbf{1}, \ \boldsymbol{z} \ \succeq \ \mathbf{0}, \ \xi_i z_i \ \le \ \mathcal{P}_{\max}. \end{aligned} \tag{4.17}$$

Again, we see that this is a convex optimization problem and can be solved by standard methods. Let us denote the optimal solution as $\left(\kappa_{p,i}^{(\mathrm{opt})}, g_{p,i}^{(\mathrm{opt})}\right)$. Similarly, we note that

Lemma 4.3.3. *Optimal solution of* $(\boldsymbol{\kappa}, \boldsymbol{z})$ *in (4.17) should satisfy either 1)* $\kappa_i > 0$ *and* $z_i > 0$*, or 2)* $\kappa_i = 0$ *and* $z_i = 0$ *for secondary user* i.

The proof is similar to that of Lemma 4.3.2 and thus omitted. With the additional constraints imposed on $\boldsymbol{\kappa}$ and $\boldsymbol{z}$, we see that in general it is difficult to obtain the closed-form solutions for $(\boldsymbol{\kappa}, \boldsymbol{z})$. Since the optimal solution of $(\boldsymbol{\kappa}, \boldsymbol{z})$ needs to be equal to 0 or greater than 0 simultaneously, we propose a heuristic suboptimal algorithm for Scenario B. Specifically, first we assign $\kappa_{\max}$ and $\mathcal{P}_{\max}$ to the secondary user with largest ρ_i. If there are remaining resources, we assign $\kappa_{\max}$ and $\mathcal{P}_{\max}$ to the secondary user with second largest ρ_i and so on until $\kappa_{\max}$ and $\mathcal{P}_{\max}$ cannot be assigned to any one secondary user. In this case, we merely utilize the solution in (4.16) to allocate (κ_i, g_i) to the secondary user with the next largest ρ_i and $\kappa_i = 0, \ g_i = 0$ to the rest of the secondary users.

Let us denote the suboptimal solution as $\left(\kappa_{p,i}^{(\mathrm{sub})}, g_{p,i}^{(\mathrm{sub})}\right)$. The detailed algorithm for Scenario B is illustrated in Algorithm 1.

Algorithm 1 *Heuristic Suboptimal Algorithm*

```
Sort $\rho_i$ in a decreasing order.
for $i = 1$ to N do
  if $c_0\kappa_{\max} + \mathcal{P}_{\max} < \bar{\mathcal{C}}$ then
    $\bar{\mathcal{C}} \leftarrow \bar{\mathcal{C}} - c_0\kappa_{\max} - \mathcal{P}_{\max}$; $\kappa_i \leftarrow \kappa_{\max}$; $z_i \leftarrow \mathcal{P}_{\max}/\xi_i$.
  else
    Compute $\kappa_i$ and $z_i$ from (4.16);
    Adjust and truncate $\kappa_i$ and $z_i$ to guarantee $\kappa_i \in (0, \kappa_{\max}]$ and $z_i \in (0, \mathcal{P}_{\max}/\xi_i]$ and stop.
  end if
end for
```

4.3.2 Case II: Optimization of Number of Samples or Amplifier Gains

In some applications, either $\boldsymbol{g}$ or $\boldsymbol{\kappa}$ may be fixed for secondary users. For example, local energy calculation may be scheduled in a fixed time slot and each secondary user is assigned same number of samples. In this case, we need to optimize the amplifier gain to achieve the desired detection probability. On the other hand, we may need to choose appropriate number of samples when the amplifier gains are fixed. Here, we first assume fixed number of samples, i.e., $\boldsymbol{\kappa} = \tilde{\boldsymbol{\kappa}}$, then we need to maximize the detection probability by choosing appropriate amplifier gains. Let us define global transmission power constraint as $\mathcal{P}_{\text{tot}} = \bar{\mathcal{C}} - c_0\mathbf{1}^{\mathrm{T}}\tilde{\boldsymbol{\kappa}}$. We now examine both these cases.

(1) Scenario A

Here, we maximize global detection probability assuming the global transmit power constraint is given as $\mathcal{P}_{\text{tot}}$. We define $a_i = \tilde{\kappa}_i \gamma_i^2$ and $b_i = \tilde{\kappa}_i \tilde{\sigma}_v^2 / |h_i|^2$. Then, the optimization problem in (4.10) is equivalent to

$$\begin{aligned} \min_{\boldsymbol{z}} \quad & \sum_{i=1}^{\mathrm{N}} \frac{a_i b_i}{z_i + b_i} \\ \text{s.t.} \quad & \boldsymbol{\xi}^{\mathrm{T}} \boldsymbol{z} \ \leq \ \mathcal{P}_{\text{tot}}, \ \boldsymbol{z} \ \succeq \ \mathbf{0}. \end{aligned} \tag{4.18}$$

It is easy to see that (4.18) is a convex optimization problem. After some manipulations, we see that the Karush-Kuhn-Tucker (KKT) conditions are

$$\frac{a_i b_i}{(z_i + b_i)^2} + u_i - \lambda_0 \xi_i \ = \ 0 \tag{4.19}$$

$$\lambda_0 (\boldsymbol{\xi}^{\mathrm{T}} \boldsymbol{z} - \mathcal{P}_{\text{tot}}) \ = \ 0 \tag{4.20}$$

$$u_i z_i \ = \ 0. \tag{4.21}$$

where $\lambda_0 \geq 0$ and $u_i \geq 0$ are Lagrangian multipliers.

First we assume that $\lambda_0 > 0$ and $u_i = 0$, then from (4.19), we see that

$$z_i \ = \ \left[\sqrt{a_i b_i / (\xi_i \lambda_0)} - b_i \right]^+, \tag{4.22}$$

where $[x]^+ = \max\{0, x\}$. Plugging this into (4.20), we have

$$\sqrt{\lambda_0} \ = \ \frac{\sum_{i \in \mathcal{S}_0} \sqrt{a_i b_i \xi_i}}{\mathcal{P}_{\text{tot}} + \sum_{i \in \mathcal{S}_0} b_i \xi_i}. \tag{4.23}$$

where $\mathcal{S}_0 = \{i | z_i > 0\}$. Then, we need to determine the set $\mathcal{S}_0$ to obtain the closed-form solution for $\boldsymbol{z}$. To do this, let us define $\beta_i = \sqrt{b_i \xi_i / a_i}$. Without loss of generality, we assume $\beta_1 \le \beta_2 \le \cdots \le \beta_{\mathrm{N}}$. After some derivations, as outlined in the Appendix C.3, we have

$$\mathcal{S}_0 = \begin{cases} \{1, \cdots, i_{\mathcal{S}} | f(i_{\mathcal{S}}) < 1, f(i_{\mathcal{S}}+1) \ge 1\}, & f(\mathrm{N}) \ge 1 \\ \{1, \cdots, \mathrm{N}\}, & \text{otherwise}, \end{cases} \tag{4.24}$$

where

$$f(i) = \frac{\beta_i \sum_{j=1}^{i} \sqrt{a_j b_j \xi_j}}{\mathcal{P}_{\mathrm{tot}} + \sum_{j=1}^{i} b_j \xi_j}. \tag{4.25}$$

Thus, plugging (4.23) into (4.22), the optimal amplifier gains can be obtained as

$$g_{p,i}^{(\mathrm{opt})} = \begin{cases} \left[\frac{\tilde{\kappa}_i \tilde{\sigma}_v^2}{|h_i|^2} \left(\frac{\gamma_i |h_i|}{\sqrt{\xi_i}} \eta - 1 \right) \right]^{1/2}, & i \in \mathcal{S}_0 \\ 0, & i \notin \mathcal{S}_0, \end{cases} \tag{4.26}$$

where $\eta = \frac{\sum_{i \in \mathcal{S}_0} \tilde{\kappa}_i \xi_i / |h_i|^2 + \mathcal{P}_{\mathrm{tot}} / \tilde{\sigma}_v^2}{\sum_{i \in \mathcal{S}_0} \tilde{\kappa}_i \sqrt{\xi_i} \gamma_i / |h_i|}$.

Remark: The optimal amplifier gains follow the water-filling strategy, i.e., with larger β_i, the chance for the secondary user to be inactive is higher, where β_i is a measure of the observation and fusion channel quality. Note that $\beta_i \propto 1/(\gamma_i |h_i|)$. Hence, when the local received SNR is low or the fusion channel quality is poor, the secondary user tends not to transmit the local calculated energy to the fusion center.

For comparison, we consider two suboptimal solutions for this optimization problem as follows:

- A simple solution is to choose equal transmission power for each secondary user, i.e., $g_{p,i}^{(\mathrm{equ})} = \sqrt{\mathcal{P}_{\mathrm{tot}}/(\mathrm{N}\xi_i)}$.

- Using the Cauchy-Schwarz inequality, we see that $\mathrm{P}_d(\kappa_\infty)$ in (4.8) can be maximized when $g_i = c\gamma_i^2|h_i|^2/\xi_i^2$, where c is a constant. Based on this, we propose an alternate suboptimal solution for amplifier gains, i.e.,

$$g_{p,i}^{(\mathrm{sub})} = \left(\frac{\gamma_i^2|h_i|^2/\xi_i^2}{\sum_{i=1}^{\mathrm{N}}\gamma_i^2|h_i|^2/\xi_i}\mathcal{P}_{\mathrm{tot}}\right)^{1/2}.$$

Let us denote the asymptotic detection probability when $\tilde{\kappa}_i \to \infty$ for these three solutions of amplifier gains as $\mathrm{P}_d^{(\mathrm{opt})}(\kappa_\infty)$, $\mathrm{P}_d^{(\mathrm{equ})}(\kappa_\infty)$ and $\mathrm{P}_d^{(\mathrm{sub})}(\kappa_\infty)$. Then, we note that

Lemma 4.3.4. *When* $\beta_2 > \beta_1$, $\mathrm{P}_d^{(\mathrm{equ})}(\kappa_\infty) \leq \mathrm{P}_d^{(\mathrm{sub})}(\kappa_\infty) \leq \mathrm{P}_d^{(\mathrm{opt})}(\kappa_\infty)$.

Proof. From Appendix C.3, we see that

$$f(2) = \frac{b_1\xi_1 + b_2\xi_2 + (\beta_2 - \beta_1)\sqrt{a_1 b_1 \xi_1}}{b_1\xi_1 + b_2\xi_2 + \mathcal{P}_{\mathrm{tot}}}.$$

As $\tilde{\kappa}_i \to \infty$, we have $a_1, b_1 \to \infty$. Additionally, since $\beta_2 > \beta_1$, then $(\beta_2 - \beta_1)\sqrt{a_1 b_1 \xi_1} > \mathcal{P}_{\mathrm{tot}}$. Thus, $f(2) > 1$ and $\mathcal{S}_0 = \{1\}$. This indicates that the optimal amplifier gains are $g_1^2 = \mathcal{P}_{\mathrm{tot}}/\xi_1$ and $g_i = 0$ $(i > 1)$. Then, we have

$$\mathrm{P}_d^{(\mathrm{opt})}(\kappa_\infty) = \mathcal{Q}\left(\mathcal{Q}^{-1}(\alpha) - \frac{1}{\tilde{\sigma}_v}\Big(\mathcal{P}_{\mathrm{tot}}\max\{\theta_i\}\Big)^{1/2}\right),$$

where $\theta_i = \gamma_i^2 |h_i|^2/\xi_i$. Furthermore, we note that

$$\begin{aligned}
\mathrm{P}_d^{(\mathrm{sub})}(\kappa_\infty) &= \mathcal{Q}\left(\mathcal{Q}^{-1}(\alpha) - \frac{1}{\tilde{\sigma}_v}\left(\mathcal{P}_{\mathrm{tot}}\|\boldsymbol{\theta}\|^2/(\mathbf{1}^{\mathrm{T}}\boldsymbol{\theta})\right)^{1/2}\right), \\
\mathrm{P}_d^{(\mathrm{equ})}(\kappa_\infty) &= \mathcal{Q}\left(\mathcal{Q}^{-1}(\alpha) - \frac{1}{\tilde{\sigma}_v}\left(\mathcal{P}_{\mathrm{tot}}(\mathbf{1}^{\mathrm{T}}\boldsymbol{\theta})/\mathrm{N}\right)^{1/2}\right),
\end{aligned}$$

where $\boldsymbol{\theta} = [\theta_1, \theta_2, \cdots, \theta_{\mathrm{N}}]^{\mathrm{T}}$. Since $\max\{\theta_i\} \cdot (\mathbf{1}^{\mathrm{T}}\boldsymbol{\theta}) \geq \|\boldsymbol{\theta}\|^2$ and $\mathrm{N} \cdot \|\boldsymbol{\theta}\|^2 \geq (\mathbf{1}^{\mathrm{T}}\boldsymbol{\theta})^2$, we can conclude the proof. □

(2) Scenario B

Next, we maximize global detection probability assuming the global transmit power constraint $\mathcal{P}_{\mathrm{tot}}$ and the individual transmit power limit $\mathcal{P}_{\max}$. In this scenario, the optimization problem in (4.11) becomes

$$\begin{aligned}
\min_{\boldsymbol{z}} \quad & \sum_{i=1}^{\mathrm{N}} \frac{a_i b_i}{z_i + b_i} \\
\text{s.t.} \quad & \boldsymbol{\xi}^{\mathrm{T}}\boldsymbol{z} \;\leq\; \mathcal{P}_{\mathrm{tot}},\; \boldsymbol{z} \;\succeq\; \mathbf{0},\; \xi_i z_i \;\leq\; \mathcal{P}_{\max}.
\end{aligned} \tag{4.27}$$

With the additional constraint in (4.27) as compared to (4.18), the updated KKT conditions can be computed as

$$\frac{a_i b_i}{(z_i + b_i)^2} + u_i - v_i\xi_i - \lambda_0\xi_i \;=\; 0 \tag{4.28}$$

$$v_i(\xi_i z_i - \mathcal{P}_{\max}) \;=\; 0, \tag{4.29}$$

where $v_i \geq 0$ are Lagrangian multipliers. First we assume that $\lambda_0 > 0$ and $u_i = v_i = 0$, then from (4.28), we see that

$$z_i = \sqrt{a_i b_i/(\xi_i \lambda_0)} - b_i.$$

Thus, based on the value of $\sqrt{\lambda_0}$, we can determine the optimal solution of z_i as

$$z_i \;=\; \begin{cases} 0, & \text{if } \sqrt{\lambda_0} > \sqrt{a_i/(b_i\xi_i)} \\ \mathcal{P}_{\max}/\xi_i, & \text{if } 0 < \sqrt{\lambda_0} < \sqrt{a_i b_i \xi_i}/(\mathcal{P}_{\max} + b_i\xi_i) \\ \sqrt{a_i b_i/(\xi_i\lambda_0)} - b_i, & \text{otherwise.} \end{cases}$$

Let us define two disjoint sets for secondary users as $\mathcal{S}_1 = \{i | z_i = \mathcal{P}_{\max}/\xi_i\}$ and $\mathcal{S}_2 = \{i | 0 < z_i < \mathcal{P}_{\max}/\xi_i\}$. Plugging z_i into (4.20), we have

$$|\mathcal{S}_1|\mathcal{P}_{\max} + \frac{1}{\sqrt{\lambda_0}} \sum_{i \in \mathcal{S}_2} \sqrt{a_i b_i \xi_i} - \sum_{i \in \mathcal{S}_2} b_i \xi_i \;=\; \mathcal{P}_{\text{tot}},$$

which implies that

$$\sqrt{\lambda_0} \;=\; \frac{\sum_{i \in \mathcal{S}_2} \sqrt{a_i b_i \xi_i}}{\mathcal{P}_{\text{tot}} - |\mathcal{S}_1|\mathcal{P}_{\max} + \sum_{i \in \mathcal{S}_2} b_i \xi_i}.$$

In order to determine $\mathcal{S}_1$, $\mathcal{S}_2$ and $\sqrt{\lambda_0}$ and thus obtain the closed-form solution for z_i, we propose a two-stage generalized water-filling algorithm as follows:

1. In the first stage, we aim to determine the set $\mathcal{S}_1$. To do this, let us define $\tilde{\beta}_i = \frac{\mathcal{P}_{\max} + b_i\xi_i}{\sqrt{a_i b_i \xi_i}}$. Without loss of generality, we assume $\tilde{\beta}_1 \leq \tilde{\beta}_2 \leq \cdots \leq \tilde{\beta}_{\text{N}}$.

Then, similar to Scenario A, $\mathcal{S}_1$ can be obtained by (4.24) with

$$\tilde{f}(i) \;=\; \frac{\tilde{\beta}_i \sum_{m\in\tilde{\mathcal{S}}_i} \sqrt{a_m b_m \xi_m}}{\mathcal{P}_{\text{tot}} - i\mathcal{P}_{\max} + \sum_{m\in\tilde{\mathcal{S}}_i} b_m \xi_m}, \quad i \leq \left\lfloor \frac{\mathcal{P}_{\text{tot}}}{\mathcal{P}_{\max}} \right\rfloor, \tag{4.30}$$

where $\tilde{\mathcal{S}}_i = \{m|\beta_m < \tilde{\beta}_i,\; i < m \leq \mathrm{N}\}$. After $\mathcal{S}_1$ is determined, we have $z_i = \mathcal{P}_{\max}/\xi_i,\; \forall i \in \mathcal{S}_1$. For an outline, please see Appendix C.4.

2. In the second stage, we follow the similar procedure in Scenario A to obtain $\mathcal{S}_2$ and z_i for $i \notin \mathcal{S}_1$. The solution is given in (4.26), except that $\mathcal{P}_{\text{tot}}$ and N are replaced by $\mathcal{P}_{\text{tot}} - |\mathcal{S}_1|\mathcal{P}_{\max}$ and $\mathrm{N} - |\mathcal{S}_1|$, respectively.

To summarize, the detailed generalized water-filling algorithm for Scenario B is illustrated in Algorithm 2. With amplifier gains fixed, we need to optimize the number of samples to achieve the desired detection probability. We note that in this case, the optimal solutions of the number of samples are similar to those of the amplifier gains in both scenarios (with additional relaxation consideration), thus omitted from this dissertation.

4.4 Optimization: Minimization of System Level Cost

In the section, we aim to minimize the system level cost of cooperative spectrum sensing to achieve a targeted detection probability. Similar to the optimization problem in Section 4.3, we consider two scenarios which depend on whether additional constraints are imposed or not. For instance, in Scenario A, the optimization

Algorithm 2 *Generalized Water-filing Algorithm*

```
Stage 1: Sort $\tilde{\beta}_i$ in an increasing order.
for $i = 1$ to $\lfloor \frac{\mathcal{P}_{\text{tot}}}{\mathcal{P}_{\max}} \rfloor$ do
    Compute $\tilde{f}(i)$ from (4.30);
    if $\tilde{f}(i) \geq 1$ then
        Set $\mathcal{S}_1 = \{1, \cdots, i\}$ and stop.
    end if
end for
for $i \in \mathcal{S}_1$ do
    $z_i \leftarrow \mathcal{P}_{\max}/\xi_i$.
end for
Stage 2: For $j \notin \mathcal{S}_1$, sort $\beta_j$ in an increasing order and set $\mathcal{P}_{\text{tot}} \leftarrow \mathcal{P}_{\text{tot}} - |\mathcal{S}_1|\mathcal{P}_{\max}$
and $\mathrm{N} \leftarrow \mathrm{N} - |\mathcal{S}_1|$.
for $j = 1$ to N do
    Compute $f(j)$ from (4.25);
    if $f(j) \geq 1$ then
        Set $\mathcal{S}_2 = \{1, \cdots, j\}$ and stop.
    end if
end for
for $j \in \mathcal{S}_2$ do
    Compute $\eta$ and $z_j$ from (4.26).
end for
```

problem can be formulated as:

$$\begin{aligned} \min_{\boldsymbol{\kappa},\boldsymbol{g}} \quad & \mathcal{C}(\boldsymbol{\kappa},\boldsymbol{g}) \\ \text{s.t.} \quad & \mathrm{P}_d(\boldsymbol{\kappa},\boldsymbol{g}) \;\geq\; \bar{\mathrm{P}}_d, \\ & \boldsymbol{\kappa} \in \mathcal{Z}_+^{\mathrm{N}},\; \boldsymbol{g} \in \mathcal{R}_+^{\mathrm{N}}, \end{aligned} \tag{4.31}$$

where $\bar{\mathrm{P}}_d$ is a predefined detection probability threshold and $\bar{\mathrm{P}}_d > \alpha$. Similar to the analysis in Section 4.3.1, we consider the relaxation, i.e., $\boldsymbol{\kappa} \in \mathcal{R}_+^{\mathrm{N}}$ to this MINLP, and the optimal solution of this relaxation problem is stated as follows:

Theorem 4.4.1. *Consider the relaxation optimization problem in (4.31) and ρ_i*

as defined in (4.14). Then,

$$\kappa_{d,i}^{(\mathrm{opt})} = \begin{cases} \frac{\epsilon}{\gamma_i^2}\Big(1+\sqrt{\frac{\xi_i}{c_0}}\frac{\tilde{\sigma}_v}{|h_i|}\Big), & i=1 \\ 0, & i>1, \end{cases}$$

$$g_{d,i}^{(\mathrm{opt})} = \begin{cases} \Big[\frac{\epsilon\tilde{\sigma}_v^2}{\gamma_i^2|h_i|^2}\Big(1+\sqrt{\frac{c_0}{\xi_i}}\frac{|h_i|}{\tilde{\sigma}_v}\Big)\Big]^{1/2}, & i=1 \\ 0, & i>1, \end{cases}$$

where $\epsilon = [\mathcal{Q}^{-1}(\alpha) - \mathcal{Q}^{-1}(\bar{\mathrm{P}}_d)]^2$.

The proof is similar to that of Theorem 4.3.1 and thus omitted. Similarly, we may consider a ceiling operation for the number of samples as a near-optimal solution for (4.31). Additionally, we see that only *one* secondary user needs be active for collecting the samples for local energy calculation and transmitting energy statistics to fusion center. We have separately examined the optimization problem for the remaining cases considered in Section 4.3, i.e., when jointly designing $\boldsymbol{\kappa}$ and $\boldsymbol{g}$ for Scenario B; and when designing either $\boldsymbol{\kappa}$ or $\boldsymbol{g}$ for both Scenario A and Scenario B.

4.5 Simulation Results

In this section, we present numerical results for the optimal design of cooperative spectrum sensing in cost constrained cognitive radio networks. In the following results, we assume $\mathrm{N} = 6$, $\sigma_n^2 = \sigma_v^2 = 1$, $c_0 = 1$, $\boldsymbol{h} = [1.56, 1.99, 0.37, 1.52, 0.39, 1.98]^{\mathsf{T}}$, $\boldsymbol{\gamma} = [-8.86, -15.23, -7.21, -5.09, -10.00, -10.97]^{\mathsf{T}}$(dB) and $\pi_0 = \pi_1 = 0.5$. Here, we define the global fusion SNR as $\mathsf{SNR} = \mathcal{P}_{\mathrm{tot}}/\sigma_v^2$.

(1) Optimization: Maximization of Detection Probability

In Fig. 4.2, we plot the receiver operating characteristic (ROC) performance for optimization in Case I (joint optimization of $\boldsymbol{\kappa}$ and $\boldsymbol{g}$) for different solutions of $(\boldsymbol{\kappa}, \boldsymbol{g})$. In this simulation, we choose $\bar{\mathcal{C}} = 20$dB and utilize standard MINLP method [68] in Scenario A and interior-point method to solve the optimization problem in Scenario B. In Scenario B, we choose $\kappa_{\max} = 0.2\lfloor \bar{\mathcal{C}}/c_0 \rfloor$ and $\mathcal{P}_{\max} = 0.2\bar{\mathcal{C}}$. For comparison, we consider equal number of samples and amplifier gains as a suboptimal solution. As illustrated in Fig. 4.2, in Scenario A, our proposed suboptimal solution in (4.16) is near optimal as previously mentioned. Furthermore, we observe that the detection performance is degraded with the additional constraints in Scenario B. Additionally, we note that our proposed suboptimal algorithm in both Scenario A and B has negligible performance loss compared to the optimal solution.

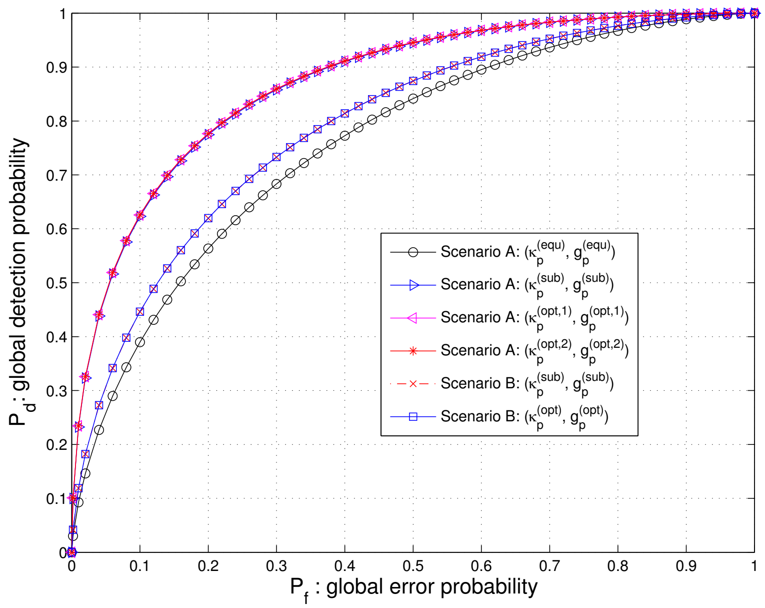


Figure 4.2: Optimization in Case I: global detection probability versus false alarm probability for different solutions of $(\boldsymbol{\kappa}, \boldsymbol{g})$.

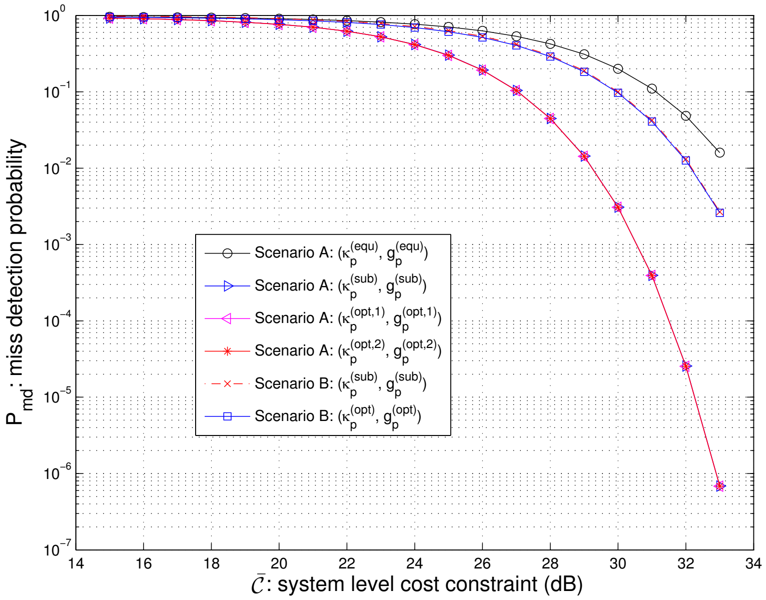


Figure 4.3: Optimization in Case I: miss detection probability for different solutions of $(\boldsymbol{\kappa}, \boldsymbol{g})$.

Fig. 4.3 shows the miss detection probability versus system level cost constraint for optimization in Case I for different solutions of $(\boldsymbol{\kappa}, \boldsymbol{g})$. In this simulation, we choose $\alpha = 0.01$, which indicates the desired probability of false alarm is 1%. As expected, we observe a similar trend as in Fig. 4.2.

Fig. 4.4 shows the miss detection probability versus total number of samples for optimization in Case II (optimization of $\boldsymbol{g}$ given $\tilde{\boldsymbol{\kappa}}$). In the simulations, we choose $\alpha = 0.01$, $\mathsf{SNR} = 25$dB and fixed number of samples $\tilde{\kappa}_i = \lfloor \kappa_{\text{tot}}/\text{N} \rfloor$. In Scenario B, we choose $\mathcal{P}_{\max} = 0.4\mathcal{P}_{\text{tot}}$. As expected, we see that the optimal solution provides superior performance to suboptimal solutions. From the plots, we also observe that with additional individual constraints, the optimal solution for Scenario B performs worse than that of Scenario A. Furthermore, when total number of samples increases, we see that the error probability approaches the asymptotic bound. In particular, $\text{P}_d^{(\text{equ})}(\kappa_\infty) \le \text{P}_d^{(\text{sub})}(\kappa_\infty) \le \text{P}_d^{(\text{opt})}(\kappa_\infty)$ as stated

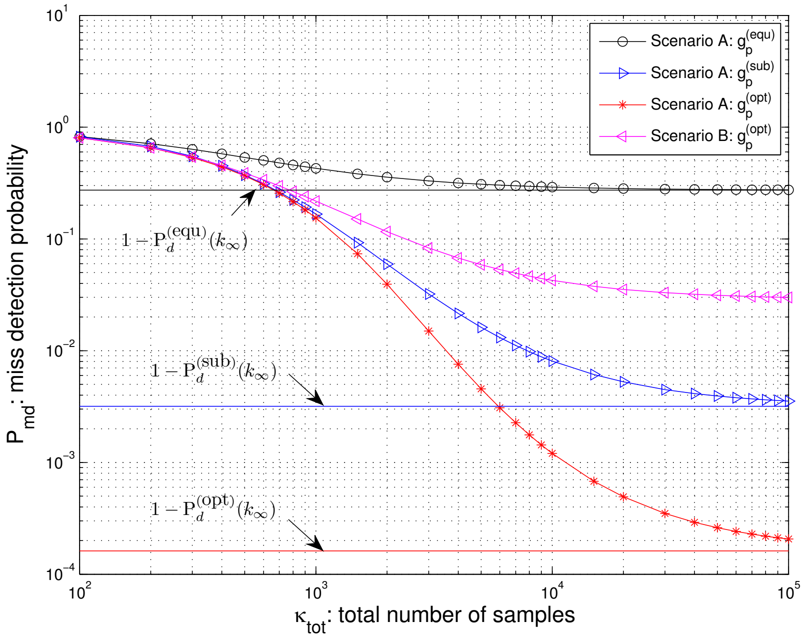


Figure 4.4: Optimization in Case II: miss detection probability for different solutions of $\boldsymbol{g}$.

in Lemma 4.3.4.

(2) Optimization: Minimization of System Level Cost

In Fig. 4.5, we plot the system level cost versus global detection probability threshold for optimization in Case I for different solutions of $(\boldsymbol{\kappa}, \boldsymbol{g})$. As expected, we see that the optimal solution in Scenario A can greatly save the system level cost compared to equal solution and optimal solution in Scenario B. For instance, 5.8dB gain can be achieved when comparing the optimal solution to the equal solution in Scenario A. From the plots, we also observe that our proposed suboptimal algorithm in both Scenario A and B has negligible performance loss compared to the optimal solution.

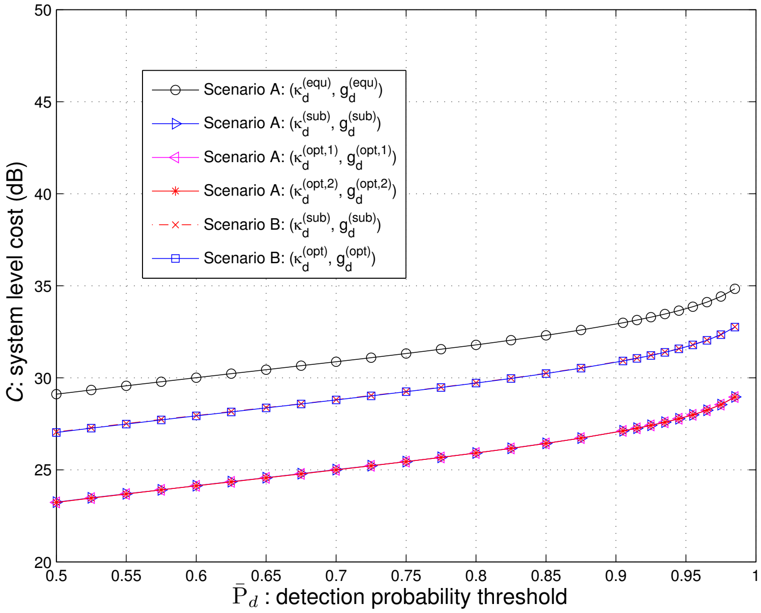


Figure 4.5: Optimization in Case I: system level cost for different solutions of $(\boldsymbol{\kappa}, \boldsymbol{g})$.

4.6 Chapter Summary

In this chapter, we have presented the optimal design for spectrum sensing in the cost constrained cognitive radio networks over the PAC scenario. Specifically, we have derived closed-form expressions for optimal solutions and proposed a generalized water-filling algorithm when number of samples or amplifier gains are fixed and additional constraints are imposed. Furthermore, when jointly designing the number of samples and amplifier gains, we have demonstrated that only *one* secondary user needs to be actively processing and transmitting local statistics to the fusion center.

Chapter 5

Cooperative Spectrum Sensing in Cost Constrained Cognitive Radio Networks: Multiple Access Channels

5.1 Introduction

In this chapter, we study the cooperative spectrum sensing in cost constrained cognitive radio network over MAC channels. Previous works [36, 69] only assumed perfect reception of hard/soft information at the fusion center, which may be inappropriate in practical settings, e.g., when cognitive radios forward their decisions to the fusion center using (unreliable) wireless links. In fact, it was shown that unreliable links between a secondary user and a fusion center can significantly impact the probability of correctly detecting the primary user when hard decision combining [70] or soft decision combining [71] is employed.

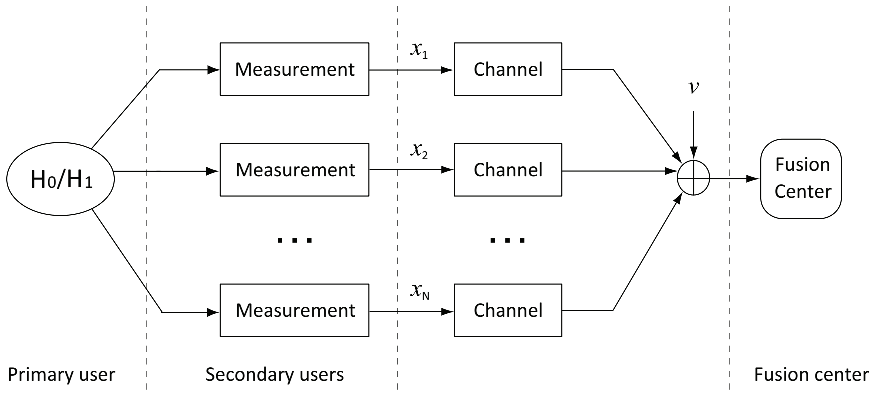


Figure 5.1: Cooperative spectrum sensing in cognitive radio networks over MAC.

As shown in Fig. 5.1, secondary users forward local measured SNR to the fusion center over MAC channels. Based on this model, we counter unreliable uplinks to the fusion center by applying beamforming amongst secondary users who communicate locally measured SNRs to a common fusion center. Under correlated lognormal shadowing, we derive optimal beamforming weights that maximize the global detection probability subject to a global transmit power constraint. We then compute the detection performance for a simplified linear array network and show that detection probability increases as the number of secondary users increases.

The remainder of this chapter is outlined as follows: Section 5.2 provides the system model; Section 5.3 presents the optimal beamforming design; Section 5.4 evaluates the detection performance. Simulation results are presented in Section 5.5.

5.2 System Model

(1) Local Measured SNR

In this chapter, we adopt the system model in [36, 69] for locally measured SNR. Specifically, we assume the logarithmic value of this SNR is given as x_i and we denote $\boldsymbol{x} = [x_1, x_2, \cdots, x_{\mathrm{N}}]^{\mathsf{T}}$. Assuming perfect local SNR estimation, the hypothesis test for $\boldsymbol{x}$ can be given as

$$\begin{cases} \mathcal{H}_0 : & \boldsymbol{x} \sim \mathcal{N}(\mu_0 \mathbf{1},\ \sigma_s^2 \mathbf{\Sigma}) \\ \mathcal{H}_1 : & \boldsymbol{x} \sim \mathcal{N}(\mu_1 \mathbf{1},\ \sigma_s^2 \mathbf{\Sigma}), \end{cases} \tag{5.1}$$

where $\mu_0 = f(R + \delta)$, $\mu_1 = f(R)$ such that $f(\cdot)$ is a distance dependent path loss function, R is the range from the TV (primary) transmitter at which secondary users are unlikely to cause interference to the primary receiver; $\delta > 0$ indicates the distance outside R; σ_s^2 is the variance of x_i; and $\mathbf{\Sigma}$ is the normalized covariance matrix of $\boldsymbol{x}$. Without loss of generality, we assume an exponential correlation model [72], i.e.,

$$[\mathbf{\Sigma}]_{ij} \;=\; \exp(-a d_{ij}), \quad i, j = 1, 2, \cdots, \mathrm{N},$$

where d_{ij} is the distance between secondary user i and j; a is a constant which depends on the environment, e.g., $a \approx 0.12$ in urban environments and $a \approx 0.002$ in suburban environments.

Using the lognormal distribution, we can first show that

$$\xi \overset{\text{def}}{=} \mathbb{E}\{x_i^2\} \;=\; \exp\left(2c^2\sigma_s^2\right)\left[\pi_0 \exp(2c\mu_0) + \pi_1 \exp(2c\mu_1)\right],$$

where $c = (\ln 10)/10$.

(2) Data Forwarding with Beamforming

We assume secondary users synchronize local SNR measurements and synchronously forward them to the fusion center using beamforming. In particular, secondary user i forwards the data x_i with beamforming weight b_i.

As shown in Fig. 5.1, the received signal at the fusion center is given as[1]

$$y \ = \ \boldsymbol{b}^{\dagger}\mathbf{H}\boldsymbol{x} + v, \tag{5.2}$$

where $\boldsymbol{b} = [b_1, b_2, \cdots, b_{\mathrm{N}}]^{\mathrm{T}}$; $\mathbf{H} = \mathrm{diag}\{h_1, h_2, \cdots, h_{\mathrm{N}}\}$, here h_i is channel gain between the secondary user i and fusion center; v is Gaussian noise, i.e., $v \sim \mathcal{N}(0, \sigma_v^2)$ and is independent of x_i. The transmit power for the secondary user i can be computed as

$$\mathcal{P}_i \ = \ b_i^2 \mathbb{E}[x_i^2] \ = \ \xi b_i^2.$$

In this chapter, we assume a global power constraint, i.e.,

$$\mathbf{1}^{\mathrm{T}}\boldsymbol{\mathcal{P}} = \mathcal{P}_{\mathrm{tot}},$$

where $\boldsymbol{\mathcal{P}} = [\mathcal{P}_1, \mathcal{P}_2, \cdots, \mathcal{P}_{\mathrm{N}}]^{\mathrm{T}}$. This is equivalent to

$$\|\boldsymbol{b}\|^2 \ = \ \mathcal{P}'_{\mathrm{tot}} \ = \ \mathcal{P}_{\mathrm{tot}}/\xi.$$

[1] We assume perfect phase synchronization among the secondary users and fusion center. The effects of imperfect phase synchronization are left for future work.

(3) NP Detection Rule

Under hypothesis $\mathcal{H}_0$ and $\mathcal{H}_1$, the received signal at the fusion center y has a Gaussian distribution, i.e.,

$$\begin{cases} \mathcal{H}_0: & y \sim \mathcal{N}\left(\mu_0 \boldsymbol{b}^\dagger \mathbf{H}\mathbf{1},\ \sigma_s^2 \boldsymbol{b}^\dagger \mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^\dagger \boldsymbol{b} + \sigma_v^2\right) \\ \mathcal{H}_1: & y \sim \mathcal{N}\left(\mu_1 \boldsymbol{b}^\dagger \mathbf{H}\mathbf{1},\ \sigma_s^2 \boldsymbol{b}^\dagger \mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^\dagger \boldsymbol{b} + \sigma_v^2\right). \end{cases} \tag{5.3}$$

Then, optimal LRT can be computed as

$$\mathcal{T}(y) \ = \ y \underset{\mathcal{H}_0}{\overset{\mathcal{H}_1}{\gtrless}} \tau_g, \tag{5.4}$$

where τ_g is the global decision threshold. After some manipulations, for NP detection with false alarm probability $\mathrm{P}_f = \alpha$, the global decision threshold can be obtained as

$$\tau_g \ = \ \mu_0 \boldsymbol{b}^\dagger \mathbf{H}\mathbf{1} + \mathcal{Q}^{-1}(\alpha)\Big(\sigma_s^2 \boldsymbol{b}^\dagger \mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^\dagger \boldsymbol{b} + \sigma_v^2\Big)^{1/2}.$$

Assuming $\Delta_\mu = \mu_1 - \mu_0$, the global detection probability is then given as

$$\mathrm{P}_d \ = \ \mathcal{Q}\Bigg(\mathcal{Q}^{-1}(\alpha) - \frac{\Delta_\mu |\boldsymbol{b}^\dagger \mathbf{H}\mathbf{1}|}{\sqrt{\sigma_s^2 \boldsymbol{b}^\dagger \mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^\dagger \boldsymbol{b} + \sigma_v^2}}\Bigg). \tag{5.5}$$

5.3 Optimal Beamforming Design

To obtain the optimal beamforming weights, we formulate an optimization problem to maximize the global detection probability in (5.5). Since $\mathcal{Q}(x)$ is a monotonically decreasing function, we see that this is equivalent to

$$\begin{aligned} \max_{\boldsymbol{b}} \quad & \frac{|\boldsymbol{b}^\dagger \mathbf{H}\mathbf{1}|^2}{\boldsymbol{b}^\dagger \mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^\dagger \boldsymbol{b} + \sigma_v^2/\sigma_s^2} \\ \text{s.t.} \quad & \|\boldsymbol{b}\|^2 \ = \ \mathcal{P}'_{\text{tot}}, \ \boldsymbol{b} \ \succeq \ \mathbf{0}. \end{aligned} \tag{5.6}$$

Using Cauchy-Schwarz inequality, we see that

$$\begin{aligned} \frac{|\boldsymbol{b}^\dagger \mathbf{H}\mathbf{1}|^2}{\boldsymbol{b}^\dagger \mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^\dagger \boldsymbol{b} + \sigma_v^2/\sigma_s^2} &= \frac{|\boldsymbol{b}^\dagger \mathbf{H}\mathbf{1}|^2}{\|(\mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^\dagger + \tilde{\gamma}\mathbf{I}_{\text{N}})^{1/2}\boldsymbol{b}\|^2} \\ &= \frac{|\tilde{\boldsymbol{b}}^\dagger (\mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^\dagger + \tilde{\gamma}\mathbf{I}_{\text{N}})^{-1/2}\mathbf{H}\mathbf{1}|^2}{\|\tilde{\boldsymbol{b}}\|^2} \\ &\leq \left\|(\mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^\dagger + \tilde{\gamma}\mathbf{I}_{\text{N}})^{-1/2}\mathbf{H}\mathbf{1}\right\|^2 \end{aligned}$$

and equality holds only when $\tilde{\boldsymbol{b}} = \eta(\mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^\dagger + \tilde{\gamma}\mathbf{I}_{\text{N}})^{-1/2}\mathbf{H}\mathbf{1}$, or equivalently, $\boldsymbol{b} = \eta(\mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^\dagger + \tilde{\gamma}\mathbf{I}_{\text{N}})^{-1}\mathbf{H}\mathbf{1}$, where η is a constant and $\tilde{\gamma} = \sigma_v^2\xi/(\sigma_s^2\mathcal{P}_{\text{tot}})$. Since $\|\boldsymbol{b}\|^2 = \mathcal{P}'_{\text{tot}}$, the optimal beamforming weights are given as

$$\boldsymbol{b}_{\text{opt}} \ = \ \sqrt{\mathcal{P}'_{\text{tot}}}\frac{(\mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^\dagger + \tilde{\gamma}\mathbf{I}_{\text{N}})^{-1}\mathbf{H}\mathbf{1}}{\left\|(\mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^\dagger + \tilde{\gamma}\mathbf{I}_{\text{N}})^{-1}\mathbf{H}\mathbf{1}\right\|}. \tag{5.7}$$

Plugging the optimal beamforming weights into (5.5), the global detection probability reduces to

$$\text{P}_d^{(\text{opt})} \ = \ \mathcal{Q}\left(\mathcal{Q}^{-1}(\alpha) - \frac{\Delta_\mu}{\sigma_s}\sqrt{\mathbf{1}^{\text{T}}\big(\boldsymbol{\Sigma} + \tilde{\gamma}(\mathbf{H}^\dagger\mathbf{H})^{-1}\big)^{-1}\mathbf{1}}\right). \tag{5.8}$$

In practice, cooperative spectrum sensing with beamforming consists of two phases: 1) a beamforming weight update phase; and 2) a sensing and reporting phase. During the first phase, the secondary users transmit training sequences to the fusion center so that the fusion center can estimate the channel gain $\mathbf{H}$. After calculating the optimal beamforming weight $\boldsymbol{b}_{\text{opt}}$ from $(5.7)^2$, the fusion center broadcasts them to the secondary users. During the second phase, the secondary users perform local spectrum measurement and then synchronously forward the data to the fusion center using $\boldsymbol{b}_{\text{opt}}$.

5.4 Performance Evaluation with Optimal Beamforming Weights

(1) Performance Evaluation

Let us denote λ_i and $\boldsymbol{v}_i$ as the eigenvalue and eigenvector of $\boldsymbol{\Sigma}$, respectively. With identical channel gains, i.e., $\mathbf{H} = h_0\mathbf{I}_{\mathrm{N}}$, we see that the global detection probability can be given as

$$\mathrm{P}_d^{(\text{opt})} \;=\; \mathcal{Q}\left(\mathcal{Q}^{-1}(\alpha) - \frac{\Delta_\mu}{\sigma_s}\left(\sum_{i=1}^{\mathrm{N}} \frac{|\mathbf{1}^{\mathrm{T}}\boldsymbol{v}_i|^2}{\lambda_i + \tilde{\gamma}/|h_0|^2}\right)^{1/2}\right). \tag{5.9}$$

In the case for independent shadowing, i.e., $\boldsymbol{\Sigma} = \mathbf{I}_{\mathrm{N}}$, we see that the global detection probability can be simplified as

$$\mathrm{P}_d^{(\text{opt})} \;=\; \mathcal{Q}\left(\mathcal{Q}^{-1}(\alpha) - \frac{\Delta_\mu}{\sigma_s}\left(\sum_{i=1}^{\mathrm{N}} \frac{|h_i|^2}{|h_i|^2 + \tilde{\gamma}}\right)^{1/2}\right). \tag{5.10}$$

[2] In this chapter, we assume that network topology and thus $\boldsymbol{\Sigma}$ are available at the fusion center.

Lemma 5.4.1. *The global detection probability in (5.9) can be bounded as*

$$\mathcal{Q}\left(\mathcal{Q}^{-1}(\alpha) - \frac{\Delta_\mu}{\sigma_s}\sqrt{\mathrm{N}/\zeta_0}\right) \;\leq\; \mathrm{P}_d^{(\mathrm{opt})} \;\leq\; \mathcal{Q}\left(\mathcal{Q}^{-1}(\alpha) - \frac{\Delta_\mu}{\sigma_s}\sqrt{\mathrm{N}/\zeta_1}\right),$$

where $\zeta_0 = \max\{\lambda_i\} + \tilde{\gamma}/|h_0|^2$ *and* $\zeta_1 = \min\{\lambda_i\} + \tilde{\gamma}/|h_0|^2$.

Proof. Let us define $\zeta_i = \lambda_i + \tilde{\gamma}/|h_0|^2$. Then, it is easy to check that

$$\sum_{i=1}^{\mathrm{N}} |\mathbf{1}^{\mathrm{T}}\boldsymbol{v}_i|^2/\zeta_i \;\leq\; \sum_{i=1}^{\mathrm{N}} |\mathbf{1}^{\mathrm{T}}\boldsymbol{v}_i|^2/\zeta_1 \;=\; \mathrm{N}/\zeta_1.$$

Similarly, we have $\sum_{i=1}^{\mathrm{N}} |\mathbf{1}^{\mathrm{T}}\boldsymbol{v}_i|^2/\zeta_i \geq \mathrm{N}/\zeta_0$. Thus, we obtain the bounds for $\mathrm{P}_d^{(\mathrm{opt})}$. This completes the proof. □

High Fusion SNR Analysis

In the high fusion SNR regime, based on the first-order Taylor series, we note that the global detection probability can be approximated as

$$\mathrm{P}_d^{(\mathrm{opt})} \;\approx\; \mathcal{Q}\left(\mathcal{Q}^{-1}(\alpha) - \frac{\Delta_\mu}{\sigma_s}\sqrt{\mathbf{1}^{\mathrm{T}}\left[\boldsymbol{\Sigma}^{-1} - \tilde{\gamma}\boldsymbol{\Sigma}^{-1}(\mathbf{H}^{\dagger}\mathbf{H})^{-1}\boldsymbol{\Sigma}^{-1}\right]\mathbf{1}}\right).$$

In particular, when $\mathcal{P}_{\mathrm{tot}} \to \infty$, then $\tilde{\gamma} \to 0$ and we have

$$\mathrm{P}_{d,\infty}^{(\mathrm{opt})} \;\overset{\mathrm{def}}{=}\; \lim_{\mathcal{P}_{\mathrm{tot}}\to\infty} \mathrm{P}_d^{(\mathrm{opt})} \;=\; \mathcal{Q}\left(\mathcal{Q}^{-1}(\alpha) - \frac{\Delta_\mu}{\sigma_s}\sqrt{\mathbf{1}^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}\mathbf{1}}\right).$$

This is equivalent to the global detection probability with perfect reception in [69].

Low Fusion SNR Analysis

In the low fusion SNR regime, we have

$$\mathrm{P}_d^{(\mathrm{opt})} \;\approx\; \mathcal{Q}\left(\mathcal{Q}^{-1}(\alpha) - \frac{\Delta_\mu}{\sigma_s}\sqrt{\mathbf{1}^{\mathrm{T}}\big[\mathbf{H}^\dagger\mathbf{H} - (1/\tilde{\gamma})\mathbf{H}^\dagger\mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^\dagger\mathbf{H}\big]\mathbf{1}/\tilde{\gamma}}\right).$$

In particular, when $\mathcal{P}_{\mathrm{tot}} \to 0$, we have

$$\mathrm{P}_{d,0}^{(\mathrm{opt})} \;\overset{\mathrm{def}}{=}\; \lim_{\mathcal{P}_{\mathrm{tot}}\to 0} \mathrm{P}_d^{(\mathrm{opt})} \;=\; \mathcal{Q}\left(\mathcal{Q}^{-1}(\alpha) - \frac{\Delta_\mu}{\sigma_s}\sqrt{\mathbf{tr}(\mathbf{H}^\dagger\mathbf{H})/\tilde{\gamma}}\right).$$

(2) Case Study: Linear Array Network with Equal Distance

As a case study, we now consider a special network scenario where N secondary users are deployed in a linear network with (equal) distance d between them. Topology of the linear network with equal distance is shown in Fig. 5.2. For such a network, the elements of the normalized covariance matrix are given as $[\boldsymbol{\Sigma}]_{ij} = \rho^{|i-j|}$, where $\rho = \exp(-ad)$.

It is shown that the inverse of $\boldsymbol{\Sigma}$ is [73]

$$\boldsymbol{\Sigma}^{-1} \;=\; \frac{1}{1-\rho^2}\begin{bmatrix} 1 & -\rho & 0 & \cdots & 0 \\ -\rho & 1+\rho^2 & -\rho & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & -\rho & 1+\rho^2 & -\rho \\ 0 & \cdots & 0 & -\rho & 1 \end{bmatrix},$$

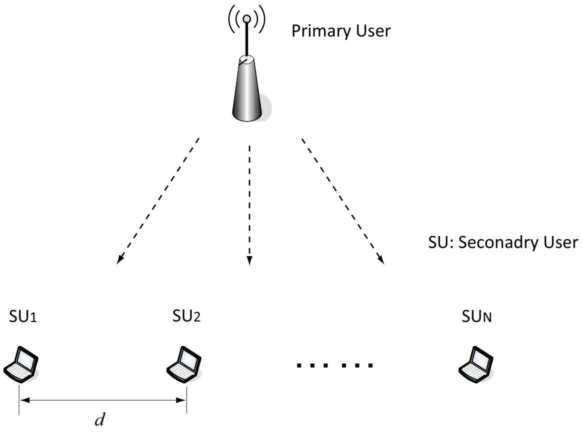


Figure 5.2: Cooperative spectrum sensing in cognitive radio networks over MAC in a linear array network with equal distance.

and the eigenvalues of $\mathbf{\Sigma}$ are

$$\lambda_i \;=\; \frac{1-\rho^2}{1-2\rho\cos\theta_i+\rho^2},\;\; i=1,\cdots,\mathrm{N},$$

where $\theta_i \in (0,\pi)$ is the ith solution of

$$\sin(\mathrm{N}+1)\theta - 2\rho\sin\mathrm{N}\theta + \rho^2\sin(\mathrm{N}-1)\theta \;=\; 0.$$

Based on this, we note that $-1 \leq \cos\theta_i \leq 1$ and then $\frac{1-\rho}{1+\rho} \leq \lambda_i \leq \frac{1+\rho}{1-\rho}$. Thus, in Lemma 5.4.1, we have

$$\zeta_0 \;=\; \frac{1+\rho}{1-\rho} + \frac{\tilde{\gamma}}{|h_0|^2} \;\text{ and }\; \zeta_1 \;=\; \frac{1-\rho}{1+\rho} + \frac{\tilde{\gamma}}{|h_0|^2}.$$

Furthermore, when $\mathcal{P}_{\mathrm{tot}} \to \infty$, we have

$$\mathrm{P}_{d,\infty}^{(\mathrm{opt})} \;=\; \mathcal{Q}\left(\mathcal{Q}^{-1}(\alpha) - \frac{\Delta_\mu}{\sigma_s}\sqrt{\frac{(1-\rho)\mathrm{N}+2\rho}{1+\rho}}\right). \tag{5.11}$$

With identical channel gains, i.e., $\mathbf{H} = h_0 \mathbf{I}_{\mathrm{N}}$, we see (following the derivation in [74]) that when N is large,

$$\mathbf{1}^{\mathrm{T}} \big(\mathbf{\Sigma} + \tilde{\gamma}_0 \mathbf{I}_{\mathrm{N}} \big)^{-1} \mathbf{1} \;=\; \frac{1-\rho}{1+\rho+\tilde{\gamma}_0(1-\rho)} \mathrm{N} + o(\mathrm{N}),$$

where $\tilde{\gamma}_0 = \tilde{\gamma}/|h_0|^2$ and $f(x) \in o(g(x))$ indicates that $\lim_{x\to\infty} f(x)/g(x) = 0$. Thus, the global detection probability can be approximated as

$$\mathrm{P}_d^{(\mathrm{opt})} \;\approx\; \mathcal{Q}\left(\mathcal{Q}^{-1}(\alpha) - \frac{\Delta_\mu}{\sigma_s} \sqrt{\frac{1-\rho}{1+\rho+\tilde{\gamma}_0(1-\rho)} \mathrm{N}} \right). \qquad (5.12)$$

From (5.12), we see that $\mathrm{P}_d^{(\mathrm{opt})}$ is an increasing function of N.

5.5 Simulation Results

In our simulations, we assume $\alpha = 0.01$, $\mu_0 = 3$, $\mu_1 = 9$, $\sigma_s = 2.3$, $\sigma_v = 1$ and $\pi_0 = \pi_1 = 0.5$. We define the global fusion SNR as $\mathsf{SNR} = \mathcal{P}_{\mathrm{tot}}/\sigma_v^2$. Here we assume a linear equi-distant array network as in Section 5.4 and measure the miss detection probability $\mathrm{P}_{md} = 1 - \mathrm{P}_d$.

Fig. 5.3 shows the miss detection probability versus global fusion SNR. In the simulations, we choose $\mathrm{N} = 5$ and $\boldsymbol{h} = [0.23 - 0.44\jmath, 0.14 - 1.08\jmath, -0.54 - 0.30\jmath, 0.51 + 0.82\jmath, 0.43 + 0.37\jmath]^{\mathrm{T}}$. From the curves, we see that the miss detection probability increases as ρ increases. Furthermore, when the global fusion SNR increases, the detection performance approaches the asymptotic bound in (5.11).

In Fig. 5.4, we plot the miss detection probability versus number of secondary

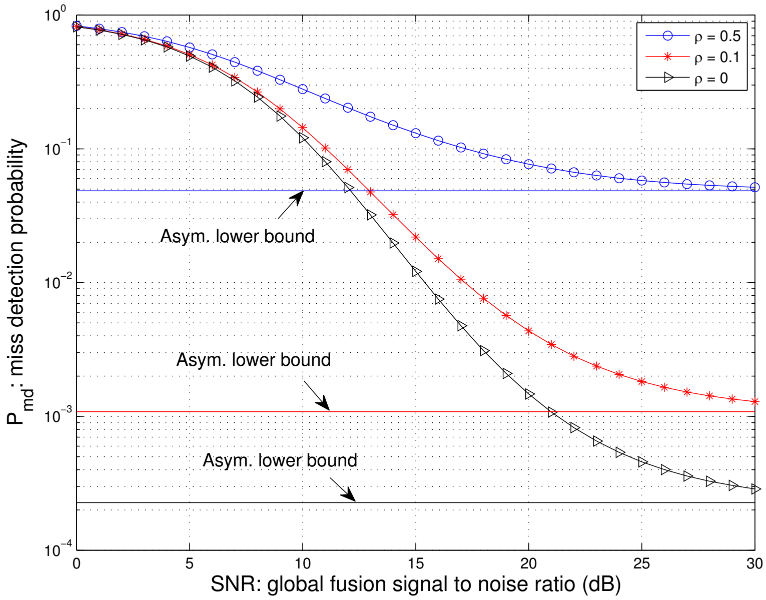


Figure 5.3: Miss detection probability versus global fusion SNR.

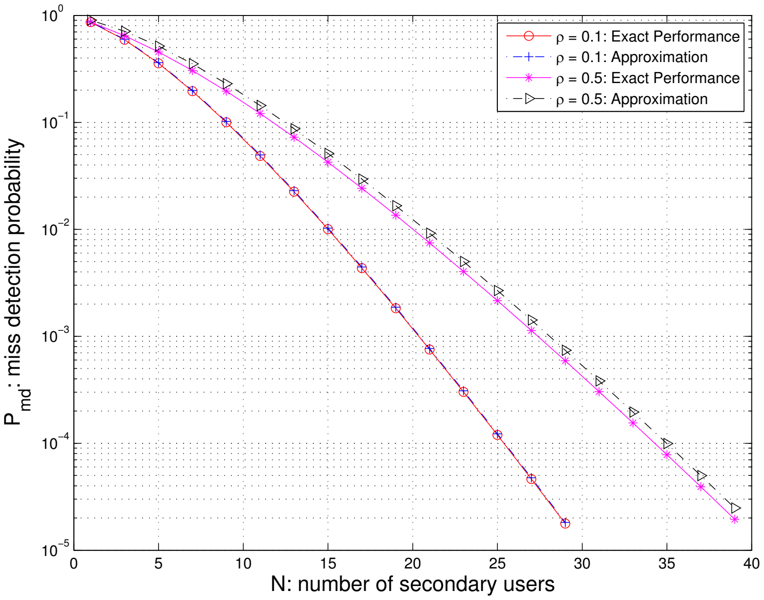


Figure 5.4: Miss detection probability versus the number of secondary users.

users. In the simulation, we choose $\mathsf{SNR} = 5\text{dB}$ and $\mathbf{H} = \mathbf{I}_{\mathrm{N}}$. As expected, we see that when more secondary users cooperatively sense the spectrum, the miss detection probability decreases. Additionally, the result in (5.12) provides a close match to the exact performance.

5.6 Chapter Summary

In this chapter, we have countered unreliable uplinks to the fusion center by applying beamforming amongst secondary users who communicate locally measured SNRs to a common fusion center. Under correlated lognormal shadowing, we have derived optimal beamforming weights that maximize the global detection probability subject to a global transmit power constraint. Then, we have computed the detection performance for a simplified linear array network and showed that detection probability increases as the number of secondary users increases.

Bibliography

[1] Federal Communications Commission, *Spectrum Policy Task Force*, Nov. 2002.

[2] M. McHenry, “Frequency Agile Spectrum Access Technologies,” *FCC Workshop Cognitive Radio*, May 2003.

[3] S. Haykin, “Cognitive Radio: Brain-Empowered Wireless Communications,” *IEEE J. Sel. Areas Commun.*, vol. 23, no. 2, pp. 201–220, Feb. 2005.

[4] IEEE 802.22 Wireless RAN, *Functional Requirements for the 802.22 WRAN Standard, IEEE 802.22 - 05/0007R46*, Oct. 2005.

[5] J. Mitola and G. Q. Maguire, “Cognitive Radios: Making Software Radios More Personal,” *IEEE Personal Commun. Mag.*, vol. 6, no. 4, pp. 13–18, Aug. 1999.

[6] K.-C. Chen, Y.-J. Peng, N. Prasad, Y.-C. Liang, and S. Sun, “Cognitive Radio Network Architecture: Part I – General Structure,” *ICUIMC ’08*, pp. 114–119, 2008.

[7] I. F. Akyildiz, W. Lee, M. C. Vuran, and S. Mohanty, “NeXt Generation/Dynamic Spectrum Access/Cognitive Radio Wireless Networks: A Survey,” *Computer Networks Journal (Elsevier)*, vol. 50, pp. 2127–2159, 2006.

[8] N. Lynch, *Distributed Algorithms.* Morgan Kaufmann Publishers, San Mateo, CA, 1996.

[9] G. Ganesan and Y. G. Li, “Cooperative Spectrum Sensing in Cognitive Radio - Part II: Multiuser Networks,” *IEEE Trans. Wireless Commun.*, vol. 6, pp. 2214–2222, June 2007.

[10] Q. Zhao and L. Tong, “Opportunistic Carrier Sensing for Energy Efficient Information Retrieval in Sensor Networks,” *EURASIP J. Wireless Commun. Netw.*, vol. 2, pp. 231–241, April. 2005.

[11] I. F. Akyildiz, W. Lee, and K. R. Chowdhurya, “CRAHNs: Cognitive Radio Ad Hoc Networks,” *Ad Hoc Networks (Elsevier)*, vol. 7, no. 5, pp. 810–836, July 2009.

[12] T. V. Krishna and A. Das, “A Survey on MAC Protocols in OSA Networks,” *Comput. Netw.*, vol. 53, no. 9, pp. 1377–1394, 2009.

[13] J. So and N. Vaidya, “Multi-Channel MAC for Ad Hoc Networks: Handling Multi-Channel Hidden Terminals Using A Single Transceiver,” *Proc. ACM MobiHoc*, pp. 222–233, May 2004.

[14] Y. R. Kondareddy and P. Agrawal, “Synchronized MAC Protocol For Multi-Hop Cognitive Radio Networks,” *ICC ’08*, pp. 3198–3202, May 2008.

[15] S. Ganeriwal, R. Kumar, and M. Srivastava, “Timingsync Protocol for Sensor Networks,” *First International Conference on Embedded Networked Sensor Systems (SenSys’03)*, pp. 138–149, 2003.

[16] J. Elson, L. Girod, and D. Estrin, "Fine-grained Network Time Synchronization Using Reference Broadcasts," *5th Symposium on Operating Systems Design and Implementation (OSDI'02)*, pp. 147–163, 2002.

[17] M. Maroti, B. Kusy, G. Simon, and A. Ldeczi, "The Flooding Time Synchronization Protocol." *2nd International Conference on Embedded Networked Sensor Systems (SenSys'04)*, pp. 39–49, 2004.

[18] F. Sivrikaya and B. Yener, "Time Synchronization in Sensor Networks: A Survey," *IEEE/ACM Trans. Netw.*, vol. 18, no. 4, pp. 45–50, July/Aug. 2004.

[19] A. Giridhar and P. R. Kumar, "Distributed Clock Synchronization over Wireless Networks: Algorithms and Analysis," *45th IEEE Conference on Decision and Control*, pp. 4915–4920, Dec. 2006.

[20] A. Scaglione and R. Pagliari, "Non-cooperative Versus Cooperative Approaches for Distributed Network Synchronization," *PerComW'07*, pp. 537–541, March 2007.

[21] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and Cooperation in Networked Multi-Agent Systems," *Proc. IEEE*, vol. 95, no. 1, pp. 215–233, Jan. 2007.

[22] L. Schenato and G. Gamba, "A Distributed Consensus Protocol for Clock Synchronization in Wireless Sensor Network," *46th IEEE Conference on Decision and Control*, pp. 2289–2294, Dec. 2006.

[23] O. Simeone and U. Spagnolini, "Distributed Time Synchronization in Wireless Sensor Networks with Coupled Discrete-Time Oscillators," *EURASIP J. Wireless Commun. and Networking*, pp. 1–13, March 2007.

[24] R. Tandra, A. Sahai, and S. M. Mishra, "What is a Spectrum Hole and What Does it Take to Recognize One?" *Proc. IEEE*, vol. 97, no. 5, pp. 824–848, May 2009.

[25] Z. Quan, S. Cui, and A. H. Sayed, "Optimal Linear Cooperation for Spectrum Sensing in Cognitive Radio Network," *IEEE J. Sel. Topics Signal Process.*, vol. 2, no. 1, pp. 28–40, Feb. 2008.

[26] J. Ma, Y. G. Li, and B. H. Juang, "Signal Processing in Cognitive Radio," *Proc. IEEE*, vol. 97, no. 5, pp. 805–823, May 2009.

[27] T. Yucek and H. Arslan, "A Survey of Spectrum Sensing Algorithms for Cognitive Radio Applications," *IEEE Commun. Surveys Tuts.*, vol. 11, no. 1, pp. 116–130, First Quarter 2009.

[28] R. Viswanathan and P. K. Varshney, "Distributed Detection with Multiple Sensors: Part I–Fundamentals," *Proc. IEEE*, vol. 85, pp. 54–63, Jan. 1997.

[29] R. S. Blum, S. A. Kassam, and H. V. Poor, "Distributed Detection with Multiple Sensors: Part II–Advanced Topics," *Proc. IEEE*, vol. 85, pp. 64–79, Jan. 1997.

[30] Z. Chair and P. K. Varshney, "Optimal Data Fusion in Multiple Sensor Detection Systems," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 22, pp. 98–101, Jan. 1986.

[31] J.-F. Chamberland and V. V. Veeravalli, "Decentralized Detection in Sensor Networks," *IEEE Trans. Signal Process.*, vol. 51, no. 2, pp. 407–416, Feb. 2003.

[32] R. Niu, B. Chen, and P. K. Varshney, “Fusion of Decisions Transmitted Over Rayleigh Fading Channels in Wireless Sensor Networks,” *IEEE Trans. Signal Process.*, vol. 54, no. 3, pp. 1018–1027, March 2006.

[33] G. Mergen and L. Tong, “Type-Based Estimation over Multiaccess Channels,” *IEEE Trans. Signal Process.*, vol. 54, no. 2, pp. 613–626, Feb. 2006.

[34] K. Liu and A. Sayeed, “Type-Based Decentralized Detection in Wireless Sensor Networks,” *IEEE Trans. Signal Process.*, vol. 55, no. 5, pp. 1899–1910, May 2007.

[35] W. Li and H. Dai, “Distributed Detection in Wireless Sensor Networks Using A Multiple Access Channel,” *IEEE Trans. Signal Process.*, vol. 55, no. 3, pp. 822–833, March 2007.

[36] E. Visotsky, S. Kuffner, and R. Peterson, “On Collaborative Detection of TV Transmissions in Support of Dynamic Spectrum Sharing,” *DySPAN, Baltimore, MD*, pp. 338–345, Nov. 2005.

[37] A. Ghasemi and E. Sousa, “Collaborative Spectrum Sensing for Opportunistic Access in Fading Environments,” *DySPAN 2005*, pp. 131–136, Nov. 2005.

[38] G. Ganesan and Y. G. Li, “Cooperative Spectrum Sensing in Cognitive Radio - Part I: Two User Networks,” *IEEE Trans. Wireless Commun.*, vol. 6, pp. 2204–2213, June 2007.

[39] J. Unnikrishnan and V. V. Veeravalli, “Cooperative Sensing for Primary Detection in Cognitive Radio,” *IEEE J. Sel. Topics Signal Process.*, vol. 2, no. 1, pp. 18–27, Feb. 2008.

[40] S. Zhang, T. Wu, and V. K. N. Lau, "A Low-Overhead Energy Detection Based Cooperative Sensing Protocol for Cognitive Radio Systems," *IEEE Trans. Wireless Commun.*, vol. 8, no. 11, pp. 5575–5561, Nov. 2009.

[41] H. G. Tanner, A. Jadbabaie, and G. J. Pappas, "Flocking in Fixed and Switching Networks," *IEEE Trans. Autom. Control*, vol. 52, no. 5, pp. 863–868, May 2007.

[42] W. Ren and E. Atkins, "Second-order Consensus Protocols in Multiple Vehicle Systems with Local Interactions," *AIAA Guidance, Navigation, and Control Conference and Exhibit*, pp. AIAA–2005–6238, Aug. 2005.

[43] P. Lin, Y. Jia, J. Du, and S. Yuan, "Distributed Consensus Control for Second-Order Agents with Fixed Topology and Time-Delay," *CCC 2007*, pp. 577–581, July 2007.

[44] W. Ren, "Second-order Consensus Algorithm with Extensions to Switching Topologies and Reference Models," *ACC '07*, pp. 1431–1436, July 2007.

[45] W. Ren, K. L. Moore, and Y. Chen, "High-Order and Model Reference Consensus Algorithms in Cooperative Control of Multi-Vehicle Systems," *ASME J. Dynamic Systems, Measurement and Control*, vol. 129, no. 5, pp. 678–688, Sept. 2007.

[46] R. A. Horn and C. R. Johnson, *Matrix Analysis.* Cambridge University Press, Cambridge, UK, 1985.

[47] R. Olfati-Saber and R. M. Murray, "Consensus Problems in Networks of Agents with Switching Topology and Time-Delays," *IEEE Trans. Autom. Control*, vol. 49, no. 9, pp. 1520–1533, Sept. 2004.

[48] L. Xiao and S. Boyd, "Fast Linear Iterations for Distributed Averaging," *42nd IEEE CDC*, vol. 5, pp. 4997–5002, Dec 2003.

[49] C. D. Meyer, *Matrix Analysis and Applied Linear Algebra.* SIAM: Society for Industrial and Applied Mathematics, 2001.

[50] G. Xiong and S. Kishore, "Second Order Distributed Consensus Time Synchronization Algorithm for Wireless Sensor Networks," *IEEE GLOBECOM 2008*, pp. 1–5, Dec. 2008.

[51] L. Xiao, S. Boyd, and S. Lall, "A Scheme for Robust Distributed Sensor Fusion Based on Average Consensus," *Fourth International Symposium on Information Processing in Sensor Networks*, pp. 63–70, April 2005.

[52] L. Xiao, S. Boyd, and S. Kim, "Distributed Average Consensus with Least-Mean-Square Deviation," *IEEE Trans. Parallel Distrib. Syst.*, vol. 67, no. 1, pp. 33–46, 2007.

[53] H. S. Abdel-Ghaffar, "Analysis of Synchronization Algorithms with Time-Out Control Over Networks with Exponentially Symmetric Delays," *IEEE Trans. Commun.*, vol. 50, no. 10, pp. 1652–1661, Oct. 2002.

[54] J. Proakis, *Digital Communications, 4th Edition.* McGraw-Hill, 2000.

[55] W. C. Lindsey, F. Ghazvinian, W. C. Hagmann, and K. Dessouky, "Network Synchronization," *Proc. IEEE*, vol. 73, no. 10, pp. 1445–1467, Oct. 1985.

[56] R. M. Gray, "Toeplitz and circulant matrices: A review," 2002.

[57] D. Spielman, *Spectral Graph Theory and its Applications.* Lecture Note, MIT.

[58] B. Mohar, “Some applications of laplace eigenvalues of graphs,” *Graph Symmetry: Algebraic Methods and Applications*, pp. 225–275, 1997.

[59] S. Appadwedula, V. V. Veeravalli, and D. L. Jones, “Energy-Efficient Detection in Sensor Networks,” *IEEE J. Sel. Areas Commun.*, vol. 23, no. 4, pp. 693–702, April 2004.

[60] Y. C. Liang, Y. Zeng, E. C. Y. Peh, and A. T. Hoang, “Sensing-Throughput Tradeoff for Cognitive Radio Networks,” *IEEE Trans. Wireless Commun.*, vol. 7, no. 4, pp. 1326–1337, Apr. 2008.

[61] Z. Quan, S. Cui, A. H. Sayed, and H. V. Poor, “Optimal Multiband Joint Detection for Spectrum Sensing in Cognitive Radio Networks,” *IEEE Trans. Signal Process.*, vol. 57, no. 3, pp. 1128–1140, Mar. 2009.

[62] A. Sahai, N. Hoven, and R. Tandra, “Some Fundamental Limits on Cognitive Radio,” *Proc. Allerton Conf. Communication, Control, and Computing*, pp. 131–136, Oct. 2004.

[63] L. A. Wolsey, *Integer Programming.* Wiley, 1998.

[64] S. Boyd and L. Vandenberghe, *Convex Optimization.* Cambridge, U.K.: Cambridge Univ. Press, 2003.

[65] R. Knopp and P. Humblet, “Information Capacity and Power Control in Single Cell Multiuser Communications,” *ICC*, pp. 331–335, June 1995.

[66] A. Bletsas, A. Khisti, D. P. Reed, and A. Lippman, “A Simple Cooperative Diversity Method Based on Network Path Selection,” *IEEE J. Sel. Areas Commun.*, vol. 24, no. 3, pp. 659–672, March 2006.

[67] T. S. Rappaport, *Wireless Communications: Principles and Practice.* Prentice Hall, Upper Saddle River, NJ, USA, 1996.

[68] I. E. Grossmann and Z. Kravanja, *Mixed-Integer Nonlinear Programming: A Survey of Algorithms and Applications*, 1997.

[69] A. Ghasemi and E. S. Sousa, "Asymptotic Performance of Collaborative Spectrum Sensing under Correlated Log-Normal Shadowing," *IEEE Commun. Lett.*, vol. 11, no. 1, pp. 34–36, Jan. 2007.

[70] W. Zhang and K. B. Letaief, "Cooperative spectrum sensing with transmit and relay diversity in cognitive radio networks," *IEEE Trans. Wireless Commun.*, vol. 7, no. 12, pp. 4761–4766, Dec. 2008.

[71] G. Xiong, S. Kishore, and A. Yener, "On Performance Evaluation of Cooperative Spectrum Sensing in Cognitive Radio Networks," *CISS 2010, Princeton, NJ*, pp. 1–6, March 2010.

[72] M. Gudmundson, "A Correlation Model for Shadow Fading in Mobile Radio," *Electron. Lett.*, vol. 27, pp. 2146–2147, Nov. 1991.

[73] U. Grenander and G. Szego, *Toeplitz Forms and Their Applications.* Chelsea Pub. Co., New York, 1984.

[74] J.-F. Chamberland and V. V. Veeravalli, "How Dense Should a Sensor Network Be for Detection With Correlated Observations?" *IEEE Trans. Inf. Theory*, vol. 52, no. 11, pp. 5099–5106, Nov. 2006.

[75] J. B. Lasserre, "A Trace Inequality for Matrix Product," *IEEE Trans. Autom. Control*, vol. 40, no. 8, pp. 1500–1501, Aug. 1995.

[76] W. Xing, Q. Zhang, and Q. Wang, “A Trace Bound for a General Square Matrix Product,” *IEEE Trans. Autom. Control*, vol. 45, no. 8, pp. 1563–1565, Aug. 2000.

[77] D. Bertsimas and J. Tsitsiklis, *Introduction to Linear Optimization.* Athena Scientific, Belmont, Massachusetts, 1997.

[78] S. Cui, J. Xiao, A. J. Goldsmith, Z.-Q. Luo, and H. V. Poor, “Estimation Diversity and Energy Efficiency in Distributed Sensing,” *IEEE Trans. Signal Process.*, vol. 55, no. 9, pp. 4683–4695, Sept. 2007.

Appendix A

Proofs for Results in Chapter 2

A.1 Solution for Convergence Region in (2.17)

Let us denote the pairs of eigenvalues of $\mathbf{H}$ corresponding to $\lambda_i(\mathbf{L})$ as $\lambda_{i'}(\mathbf{H})$ and $\lambda_{i''}(\mathbf{H})$, i.e.,

$$\begin{cases} \lambda_{i'}(\mathbf{H}) &= \frac{1}{2}\left[1-\varepsilon\lambda_i(\mathbf{L})+\sqrt{(1-\varepsilon\lambda_i(\mathbf{L}))^2+4\gamma\varepsilon\lambda_i(\mathbf{L})}\right] \\ \lambda_{i''}(\mathbf{H}) &= \frac{1}{2}\left[1-\varepsilon\lambda_i(\mathbf{L})-\sqrt{(1-\varepsilon\lambda_i(\mathbf{L}))^2+4\gamma\varepsilon\lambda_i(\mathbf{L})}\right]. \end{cases}$$

Now, we examine the convergence region for the second-order DCTS algorithm based on the conditions $|\lambda_{i'}(\mathbf{H})| < 1, 1 < i' \leq \mathrm{N}$ and $|\lambda_{i''}(\mathbf{H})| < 1, 1 < i'' \leq \mathrm{N}$.

Case I: When $\lambda_{i'}(\mathbf{H})$ and $\lambda_{i''}(\mathbf{H})$ are real values

In this case, we have $(1 - \varepsilon\lambda_i(\mathbf{L}))^2 + 4\gamma\varepsilon\lambda_i(\mathbf{L}) \geq 0$, i.e.,

$$\gamma \;\geq\; -\frac{[1 - \varepsilon\lambda_i(\mathbf{L})]^2}{4\varepsilon\lambda_i(\mathbf{L})},\; 1 < i \leq \mathrm{N}. \tag{A.1}$$

In the following, we assume that $1 < i, i', i'' \leq \mathrm{N}$ unless otherwise stated.

1. First, we consider the convergence region for $\lambda_{i'}(\mathbf{H})$. After some manipulations, we can show that the convergence region is

$$\left\{\gamma < 1,\; 0 < \varepsilon < \frac{3}{\lambda_i(\mathbf{L})}\right\} \;\cup\; \left\{\frac{2 - \varepsilon\lambda_i(\mathbf{L})}{\varepsilon\lambda_i(\mathbf{L})} < \gamma < 1,\; \varepsilon > \frac{3}{\lambda_i(\mathbf{L})}\right\}.$$

2. Then, we consider the convergence region for $|\lambda_{i''}(\mathbf{H})| < 1$ which is given as

$$\left\{\gamma < \frac{2 - \varepsilon\lambda_i(\mathbf{L})}{\varepsilon\lambda_i(\mathbf{L})},\quad 0 < \varepsilon < \frac{3}{\lambda_i(\mathbf{L})}\right\}.$$

Combining the convergence region for $\lambda_{i'}(\mathbf{H})$ and $\lambda_{i''}(\mathbf{H})$ with (A.1), the convergence region $\mathcal{R}_1$ for this case is

$$\begin{aligned}\mathcal{R}_1 \;=\; & \left\{-\frac{[1 - \varepsilon\lambda_i(\mathbf{L})]^2}{4\varepsilon\lambda_i(\mathbf{L})} \leq \gamma < 1,\; 0 < \varepsilon < \frac{1}{\lambda_i(\mathbf{L})}\right\} \\ & \cup \left\{-\frac{[1 - \varepsilon\lambda_i(\mathbf{L})]^2}{4\varepsilon\lambda_i(\mathbf{L})} \leq \gamma < \frac{2 - \varepsilon\lambda_i(\mathbf{L})}{\varepsilon\lambda_i(\mathbf{L})}, \frac{1}{\lambda_i(\mathbf{L})} \leq \varepsilon < \frac{3}{\lambda_i(\mathbf{L})}\right\}.\end{aligned} \tag{A.2}$$

Case II: When $\lambda_{i'}(\mathbf{H})$ and $\lambda_{i''}(\mathbf{H})$ are complex values

In this case, we have $(1-\varepsilon\lambda_i(\mathbf{L}))^2 + 4\gamma\varepsilon\lambda_i(\mathbf{L}) < 0$, i.e.,

$$\gamma \;<\; -\frac{[1-\varepsilon\lambda_i(\mathbf{L})]^2}{4\varepsilon\lambda_i(\mathbf{L})}.$$

Here, $\Re\{\lambda_{i'}(\mathbf{H})\} = \Re\{\lambda_{i''}(\mathbf{H})\}$ and $\Im\{\lambda_{i'}(\mathbf{H})\} = -\Im\{\lambda_{i''}(\mathbf{H})\}$. Thus we only need to examine the convergence region for $|\lambda_{i'}(\mathbf{H})|$. In order to satisfy the conditions, we have

1. The real part of $\lambda_{i'}(\mathbf{H})$ should be less than 1, i.e., $|\Re\left\{\lambda_{i'}(\mathbf{H})\right\}| < 1$, then we have
$$0 \;<\; \varepsilon \;<\; \frac{3}{\lambda_i(\mathbf{L})}.$$
2. The imaginary part of $\lambda_{i'}(\mathbf{H})$ should be less than 1, i.e., $|\Im\left\{\lambda_{i'}(\mathbf{H})\right\}| < 1$, then we have
$$-\frac{4+[1-\varepsilon\lambda_i(\mathbf{L})]^2}{4\varepsilon\lambda_i(\mathbf{L})} \;<\; \gamma \;<\; -\frac{[1-\varepsilon\lambda_i(\mathbf{L})]^2}{4\varepsilon\lambda_i(\mathbf{L})}.$$
3. The absolute value of $\lambda_{i'}(\mathbf{H})$ should be less than 1, i.e., $\Re^2\left\{\lambda_{i'}(\mathbf{H})\right\} + \Im^2\left\{\lambda_{i'}(\mathbf{H})\right\} < 1$, then we have
$$\gamma \;>\; -\frac{1}{\varepsilon\lambda_i(\mathbf{L})}.$$

Combining the above results, the convergence region $\mathcal{R}_2$ for this case is

$$\mathcal{R}_2 \;=\; \left\{-\frac{1}{\varepsilon\lambda_i(\mathbf{L})} < \gamma < -\frac{[1-\varepsilon\lambda_i(\mathbf{L})]^2}{4\varepsilon\lambda_i(\mathbf{L})},\;\; 0 < \varepsilon < \frac{3}{\lambda_i(\mathbf{L})}\right\}. \tag{A.3}$$

By taking the union of $\mathcal{R}_1$ in (A.2) and $\mathcal{R}_2$ in (A.3) and considering the increasing order of $\lambda_i(\mathbf{L})$, the convergence region for the second-order DCTS algorithm in (2.19) is obtained.

A.2 Solution for Spectral Radius Minimization in (2.20)

Here we give a sketch solution to the spectral radius minimization problem in (2.20). Since $\lambda_2(\mathbf{L}) \leq \cdots \leq \lambda_{\mathrm{N}}(\mathbf{L})$, the optimization problem is equivalent to minimizing

$$\max \left\{ |\lambda_{2'}(\mathbf{H})|, |\lambda_{2''}(\mathbf{H})|, |\lambda_{\mathrm{N}'}(\mathbf{H})|, |\lambda_{\mathrm{N}''}(\mathbf{H})| \right\}. \tag{A.4}$$

1. First, we find the optimal γ given ε to minimize (A.4). Here we consider four different cases depending on whether $\lambda_{2'}(\mathbf{H}), \lambda_{2''}(\mathbf{H}), \lambda_{\mathrm{N}'}(\mathbf{H}), \lambda_{\mathrm{N}''}(\mathbf{H})$ are real values or complex values. After algebraic derivations, we can show that the minimum of (A.4) given ε can be achieved when $\lambda_{2'}(\mathbf{H})$ and $\lambda_{2''}(\mathbf{H})$ are real values and $\lambda_{\mathrm{N}'}(\mathbf{H})$ and $\lambda_{\mathrm{N}''}(\mathbf{H})$ are complex values. Additionally, the following equation should be satisfied:

$$|\lambda_{2'}(\mathbf{H})| \;=\; |\lambda_{\mathrm{N}'}(\mathbf{H})| \;=\; |\lambda_{\mathrm{N}''}(\mathbf{H})|.$$

   Thus, we have

$$\gamma \;=\; -\frac{\lambda_{\mathrm{N}}(\mathbf{L})\left[1-\varepsilon\lambda_2(\mathbf{L})\right]^2}{\varepsilon\left[\lambda_2(\mathbf{L})+\lambda_{\mathrm{N}}(\mathbf{L})\right]^2}. \tag{A.5}$$

2. Next, we find the optimal ε given γ to minimize (A.4). Again, this can be

achieved by taking $\Im\{\lambda_{\mathrm{N}'}(\mathbf{H})\} = 0$. Then, we have the following relationship between ε and γ:

$$\gamma \;=\; -\frac{[1-\varepsilon\lambda_{\mathrm{N}}(\mathbf{L})]^2}{4\varepsilon\lambda_{\mathrm{N}}(\mathbf{L})}. \tag{A.6}$$

Combining (A.5) with (A.6), we get (2.21).

Appendix B

Proofs for Results in Chapter 3

B.1 Proof of Lemma 3.3.3

Proof. Define $\tilde{\boldsymbol{\delta}}(k) = \boldsymbol{\delta}(k) - \mathbb{E}[\boldsymbol{\delta}(k)]$. Then, the dynamics of this vector are given as follows

$$\tilde{\boldsymbol{\delta}}(k) \;=\; \tilde{\mathbf{P}}\tilde{\boldsymbol{\delta}}(k-1) + \varepsilon\tilde{\mathbf{Q}}\mathbf{A}\boldsymbol{v}(k-1). \tag{B.1}$$

To prove this lemma, we consider instead the evolution of the covariance matrix of the disagreement vector $\boldsymbol{\Sigma}_\delta(k)$ since

$$\mathbb{E}\left[\tilde{\boldsymbol{\delta}}(k)^{\mathtt{T}}\tilde{\boldsymbol{\delta}}(k)\right] \;=\; \mathbf{tr}\left[\boldsymbol{\Sigma}_\delta(k)\right] \;=\; \mathbf{tr}\left\{\mathbb{E}\left[\tilde{\boldsymbol{\delta}}(k)\tilde{\boldsymbol{\delta}}(k)^{\mathtt{T}}\right]\right\}.$$

Then, the proof of the lemma is equivalent to proving the following statement:

$$\boldsymbol{\Sigma}_\delta(k) \;=\; \tilde{\mathbf{P}}^k\boldsymbol{\delta}(0)\boldsymbol{\delta}(0)^{\mathtt{T}}\tilde{\mathbf{P}}^k + \varepsilon^2\sigma^2\sum_{l=0}^{k-1}\tilde{\mathbf{P}}^l\tilde{\mathbf{Q}}\mathbf{A}^2\tilde{\mathbf{Q}}\tilde{\mathbf{P}}^l. \tag{B.2}$$

It is straightforward to show that (B.2) holds when $k = 1$. Now let us assume that (B.2) is true when $k = m$ $(m > 1)$, i.e.,

$$\mathbf{\Sigma}_{\delta}(m) \;=\; \tilde{\mathbf{P}}^{m}\boldsymbol{\delta}(0)\boldsymbol{\delta}(0)^{\mathtt{T}}\tilde{\mathbf{P}}^{m} + \varepsilon^{2}\sigma^{2}\sum_{l=0}^{m-1}\tilde{\mathbf{P}}^{l}\tilde{\mathbf{Q}}\mathbf{A}^{2}\tilde{\mathbf{Q}}\tilde{\mathbf{P}}^{l}.$$

When $k = m + 1$, we have

$$\begin{aligned}
\mathbf{\Sigma}_{\delta}(m+1) \;&=\; \mathbb{E}\left\{\left[\tilde{\mathbf{P}}\tilde{\boldsymbol{\delta}}(m) + \varepsilon\tilde{\mathbf{Q}}\mathbf{A}\boldsymbol{v}(m)\right]\left[\tilde{\mathbf{P}}\tilde{\boldsymbol{\delta}}(m) + \varepsilon\tilde{\mathbf{Q}}\mathbf{A}\boldsymbol{v}(m)\right]^{\mathtt{T}}\right\} \\
&=\; \tilde{\mathbf{P}}^{m+1}\boldsymbol{\delta}(0)\boldsymbol{\delta}(0)^{\mathtt{T}}\tilde{\mathbf{P}}^{m+1} + \varepsilon^{2}\sigma^{2}\sum_{l=1}^{m}\tilde{\mathbf{P}}^{l}\tilde{\mathbf{Q}}\mathbf{A}^{2}\tilde{\mathbf{Q}}\tilde{\mathbf{P}}^{l} + \varepsilon^{2}\sigma^{2}\tilde{\mathbf{Q}}\mathbf{A}^{2}\tilde{\mathbf{Q}} \\
&=\; \tilde{\mathbf{P}}^{m+1}\boldsymbol{\delta}(0)\boldsymbol{\delta}(0)^{\mathtt{T}}\tilde{\mathbf{P}}^{m+1} + \varepsilon^{2}\sigma^{2}\sum_{l=0}^{m}\tilde{\mathbf{P}}^{l}\tilde{\mathbf{Q}}\mathbf{A}^{2}\tilde{\mathbf{Q}}\tilde{\mathbf{P}}^{l}. \qquad \text{(B.3)}
\end{aligned}$$

Therefore, $\mathbf{\Sigma}_{\delta}(m+1)$ has the same form as (B.2) for $k = m + 1$. Thus, (3.11) is valid and we conclude the proof. □

B.2 Proof of Theorem 3.3.2

Before proving the theorem, first we present some known results.

Theorem B.2.1. *For any matrix* $\mathbf{A}_1$ *and any symmetric matrix* $\mathbf{A}_2$*, let* $\bar{\mathbf{A}}_1 = (\mathbf{A}_1 + \mathbf{A}_1^{\mathtt{T}})/2$*, then we have* [75]

$$\sum_{i=1}^{\mathrm{N}}\lambda_{\mathrm{N}-i+1}(\bar{\mathbf{A}}_1)\lambda_i(\mathbf{A}_2) \;\leq\; \mathbf{tr}(\mathbf{A}_1\mathbf{A}_2) \;\leq\; \sum_{i=1}^{\mathrm{N}}\lambda_i(\bar{\mathbf{A}}_1)\lambda_i(\mathbf{A}_2),$$

where $\lambda_i(\cdot)$ *denotes the ith smallest eigenvalue of a matrix. In particular, if* $\mathbf{A}_2$ *is* *a positive semidefinite matrix, we have*

$$\lambda_1(\bar{\mathbf{A}}_1)\mathbf{tr}(\mathbf{A}_2) \ \leq\ \mathbf{tr}(\mathbf{A}_1\mathbf{A}_2) \ \leq\ \lambda_{\mathrm{N}}(\bar{\mathbf{A}}_1)\mathbf{tr}(\mathbf{A}_2). \tag{B.4}$$

If $\mathbf{A}_1$ is a positive semidefinite matrix, replacing $\mathbf{A}_1$ with $\mathbf{A}_2$ in (B.4), we have [76]

$$\lambda_1(\bar{\mathbf{A}}_2)\mathbf{tr}(\mathbf{A}_1) \ \leq\ \mathbf{tr}(\mathbf{A}_1\mathbf{A}_2) \ \leq\ \lambda_{\mathrm{N}}(\bar{\mathbf{A}}_2)\mathbf{tr}(\mathbf{A}_1). \tag{B.5}$$

Combining (B.4) with (B.5), we have the following theorem:

Theorem B.2.2. *If $\mathbf{A}_1$ and $\mathbf{A}_2$ are two positive semidefinite matrices, we have*

$$\begin{aligned}\mathbf{tr}(\mathbf{A}_1\mathbf{A}_2) \ &\geq\ \max\left\{\lambda_1(\mathbf{A}_1)\mathbf{tr}(\mathbf{A}_2), \lambda_1(\mathbf{A}_2)\mathbf{tr}(\mathbf{A}_1)\right\} \\ \mathbf{tr}(\mathbf{A}_1\mathbf{A}_2) \ &\leq\ \min\left\{\lambda_{\mathrm{N}}(\mathbf{A}_1)\mathbf{tr}(\mathbf{A}_2), \lambda_{\mathrm{N}}(\mathbf{A}_2)\mathbf{tr}(\mathbf{A}_1)\right\}.\end{aligned} \tag{B.6}$$

We can now prove Theorem 3.3.2.

Proof. We know that the eigenvalues of $(\mathbf{L}+\mathbf{K})^{-2}$ are 1 and $1/\lambda_i^2(\mathbf{L}),\ i = 2, \cdots, \mathrm{N}$. Also, $\lambda_{\max}(\tilde{\mathbf{Q}}) = 1$ and $\lambda_{\min}(\tilde{\mathbf{Q}}) = 0$. Recall that

$$\lambda_i(\mathbf{W}_2) \ = \ \frac{1}{2}\left[\frac{1}{\varepsilon\lambda_i(\mathbf{L})} + \frac{1}{2-\varepsilon\lambda_i(\mathbf{L})}\right],\ i = 2, \cdots, \mathrm{N}.$$

Since $\varepsilon \in (0, 2/\lambda_{\mathrm{N}}(\mathbf{L}))$, the eigenvalues of $\mathbf{W}_2$ are nonnegative. Thus, $\lambda_{\min}(\mathbf{W}_2) = 0$. In addition, $\mathbf{W}_2$ and $\mathbf{A}^2$ are positive semidefinite matrices with $\mathbf{tr}(\mathbf{A}^2) = \mathcal{D}_{\mathrm{N}}$. For a time delay unbalanced network, $\tilde{\mathbf{Q}}\boldsymbol{u} \neq \mathbf{0}$. Based on (3.17) and (B.6), $\sigma^2_{\Delta t}$ is upper bounded by

$$\begin{aligned}
\sigma_{\Delta t}^2 &\leq \frac{\|\boldsymbol{u}\tilde{\mathbf{Q}}\|^2}{\min\{\lambda_2^2(\mathbf{L}), 1\}} + \varepsilon^2\sigma^2 \mathbf{tr}\left(\mathbf{W}_2\mathbf{A}^2\right) \\
&\leq \frac{\boldsymbol{u}^{\mathrm{T}}\tilde{\mathbf{Q}}\boldsymbol{u}}{\alpha_2} + \varepsilon^2\sigma^2 \min\left\{\lambda_{\max}(\mathbf{W}_2)\mathbf{tr}(\mathbf{A}^2), \lambda_{\max}(\mathbf{A}^2)\mathbf{tr}(\mathbf{W}_2)\right\} \\
&\leq \frac{\|\boldsymbol{u}\|^2}{\alpha_2} + \varepsilon\sigma^2 \min\left\{\mathcal{D}_{\mathrm{N}} \max\left\{\lambda_i\right\}, \lambda_{\max}(\mathbf{A}^2)\sum_{i=2}^{\mathrm{N}}\lambda_i\right\}. \qquad \text{(B.7)}
\end{aligned}$$

From [58], we know that $\lambda_{\mathrm{N}}(\mathbf{L}) \geq \frac{\mathrm{N}}{\mathrm{N}-1}\max\{d_i\} > \max\{d_i\} > 1, \ \forall i \in \mathcal{V}$. Then, $\sigma_{\Delta t}^2$ is lower bounded by

$$\begin{aligned}
\sigma_{\Delta t}^2 &\geq \frac{\|\boldsymbol{u}\tilde{\mathbf{Q}}\|^2}{\max\{\lambda_{\mathrm{N}}^2(\mathbf{L}), 1\}} + \varepsilon^2\sigma^2 \mathbf{tr}\left(\mathbf{W}_2\mathbf{A}^2\right) \\
&\geq \frac{\boldsymbol{u}^{\mathrm{T}}\tilde{\mathbf{Q}}\boldsymbol{u}}{\alpha_1} + \varepsilon^2\sigma^2 \max\left\{\lambda_{\min}(\mathbf{W}_2)\mathbf{tr}(\mathbf{A}^2), \lambda_{\min}(\mathbf{A}^2)\mathbf{tr}(\mathbf{W}_2)\right\} \\
&= \frac{\boldsymbol{u}^{\mathrm{T}}\tilde{\mathbf{Q}}\boldsymbol{u}}{\alpha_1} + \varepsilon\sigma^2\lambda_{\min}(\mathbf{A}^2)\sum_{i=2}^{\mathrm{N}}\lambda_i. \qquad \text{(B.8)}
\end{aligned}$$

This completes the proof. □

Appendix C

Proofs for Results in Chapter 4

C.1 Proof of Lemma 4.3.2

Proof. Here we prove this lemma by contradiction. First we assume that $(\boldsymbol{\kappa}, \boldsymbol{z})$ with $\kappa_i = 0, z_i > 0$ or $\kappa_i > 0, z_i = 0$ for secondary user i is the optimal solution for (4.13). Let us define the optimal value is p^*. Since $\kappa_i z_i = 0$, the objective function remains unchanged in (4.13). Then, the optimization problem becomes

$$\begin{aligned} \max_{\boldsymbol{\kappa}, \boldsymbol{z}} \quad & \sum_{j=1, j\neq i}^{\mathrm{N}} \mathcal{F}_j(\kappa_j, z_j) \\ \text{s.t.} \quad & c_0 \sum_{j=1, j\neq i}^{\mathrm{N}} \kappa_j + \sum_{j=1, j\neq i}^{\mathrm{N}} \xi_j z_j \ \leq \ \bar{\mathcal{C}}' \\ & \kappa_j \geq 0, \ z_j \geq 0, \ \ \forall j \neq i. \end{aligned} \tag{C.1}$$

where $\bar{\mathcal{C}}' = \bar{\mathcal{C}} - \xi_i z_i$ when $\kappa_i = 0, z_i > 0$, or $\bar{\mathcal{C}}' = \bar{\mathcal{C}} - c_0\kappa_i$ when $\kappa_i > 0, z_i = 0$. In either case, we see that $\bar{\mathcal{C}}' < \bar{\mathcal{C}}$. To prove this lemma, we need to find a substitute solution $(\boldsymbol{\kappa}', \boldsymbol{z}')$ with optimal value $p'^* > p^*$. To do this, let us replace the solution for secondary user i as $\kappa_i' = z_i' = 0$. In this case, the optimization problem becomes

$$\begin{aligned} \max_{\boldsymbol{\kappa},\boldsymbol{z}} \quad & \sum_{j=1,j\neq i}^{\mathrm{N}} \mathcal{F}_j(\kappa_j, z_j) \\ \text{s.t.} \quad & c_0 \sum_{j=1,j\neq i}^{\mathrm{N}} \kappa_j + \sum_{j=1,j\neq i}^{\mathrm{N}} \xi_j z_j \ \leq \ \bar{\mathcal{C}} \\ & \kappa_j \geq 0,\ z_j \geq 0,\ \ \forall j \neq i. \end{aligned} \tag{C.2}$$

Then, we see that it is equivalent to proving that the optimal value p'^* in (C.2) is greater than p^* in (C.1). Since the objective and constraint functions in these two optimization problems are identical, this can be easily proved by convex relaxation in optimization problem, which implies that we can find a substitute solution $(\boldsymbol{\kappa}', \boldsymbol{z}')$, i.e., $p'^* > p^*$. This contradicts the assumption that $(\boldsymbol{\kappa}, \boldsymbol{z})$ is the optimal solution and we can conclude the proof. □

C.2 Proof of Theorem 4.3.1

Proof. The Lagrangian function of (4.13) can be given as

$$\mathcal{L}(\boldsymbol{\kappa}, \boldsymbol{z}, \lambda_0, \boldsymbol{u}, \boldsymbol{v}) = -\sum_{i=1}^{\mathrm{N}} \frac{\kappa_i z_i}{p_i\kappa_i + q_i z_i} + \lambda_0(c_0 \mathbf{1}^{\mathrm{T}}\boldsymbol{\kappa} + \boldsymbol{\xi}^{\mathrm{T}}\boldsymbol{z}) - \boldsymbol{u}^{\mathrm{T}}\boldsymbol{\kappa} - \boldsymbol{v}^{\mathrm{T}}\boldsymbol{z} - \lambda_0\bar{\mathcal{C}},$$

where $\lambda_0 \geq 0$, $\boldsymbol{u} = [u_1, u_2, \cdots, u_{\mathrm{N}}]^{\mathrm{T}}$ and $\boldsymbol{v} = [v_1, v_2, \cdots, v_{\mathrm{N}}]^{\mathrm{T}}$, here $u_i \geq 0$ and $v_i \geq 0$ are Lagrangian multipliers. After some manipulations, the KKT conditions are

$$\frac{q_i z_i^2}{(p_i\kappa_i + q_i z_i)^2} + u_i - c_0\lambda_0 \;=\; 0 \tag{C.3}$$

$$\frac{p_i \kappa_i^2}{(p_i\kappa_i + q_i z_i)^2} + v_i - \xi_i\lambda_0 \;=\; 0 \tag{C.4}$$

$$\lambda_0 \left(c_0 \mathbf{1}^{\mathtt{T}}\boldsymbol{\kappa} + \boldsymbol{\xi}^{\mathtt{T}}\boldsymbol{z} - \bar{\mathcal{C}}\right) \;=\; 0 \tag{C.5}$$

$$u_i\kappa_i \;=\; 0, \quad v_i z_i \;=\; 0. \tag{C.6}$$

From Lemma 4.3.2, we see that u_i and v_i need to be 0 or greater than 0 simultaneously. First we assume $u_i = v_i = 0$ and $\lambda_0 > 0$, which indicates that $\kappa_i > 0$ and $z_i > 0$. Then from (C.3) and (C.4), we have $z_i = \omega_i\kappa_i$, where $\omega_i = \sqrt{c_0 p_i/(q_i\xi_i)}$. Plugging this into (4.13), the original optimization problem becomes

$$\begin{aligned} \max_{\boldsymbol{\kappa}} \quad & \sum_{i\in\mathcal{I}} s_{1i}\kappa_i \\ \text{s.t.} \quad & \sum_{i\in\mathcal{I}} s_{2i}\kappa_i \leq \bar{\mathcal{C}}, \;\; \kappa_i \geq 0, \;\; \forall i \in \mathcal{I}, \end{aligned} \tag{C.7}$$

where $\mathcal{I} = \{i|\kappa_i > 0, z_i > 0\}$, $s_{1i} = (q_i + p_i/\omega_i)^{-1}$ and $s_{2i} = c_0 + \xi_i\omega_i$. Let us define $\boldsymbol{s}_1 = [s_{11}, s_{12}, \cdots, s_{1\mathrm{N}}]^{\mathtt{T}}$ and $\boldsymbol{s}_2 = [s_{21}, s_{22}, \cdots, s_{2\mathrm{N}}]^{\mathtt{T}}$. Since adding zero will not change the objective function and constraints in (C.7), we can rewrite this optimization problem as

$$\begin{aligned} \max_{\boldsymbol{\kappa}} \quad & \boldsymbol{s}_1^{\mathtt{T}}\boldsymbol{\kappa} \\ \text{s.t.} \quad & \boldsymbol{s}_2^{\mathtt{T}}\boldsymbol{\kappa} \leq \bar{\mathcal{C}}, \;\; \boldsymbol{\kappa} \succeq \mathbf{0}. \end{aligned} \tag{C.8}$$

This is a classic linear optimization problem, thus we can solve this easily. Since the vertices of the polyhedron are the basic feasible solution for linear optimization problem [77], the optimal solution of (C.8) suggests that only one of κ_i is non-zero while others are all zero. Let us define $\rho_i = s_{1i}/s_{2i}$ and assume $\rho_1 \geq \rho_2 \geq \cdots \geq \rho_{\mathrm{N}}$. Then, the optimal solution of $(\boldsymbol{\kappa}, \boldsymbol{g})$ can be given in Theorem 4.3.1. This completes the proof. □

C.3 Solution for Set $\mathcal{S}_0$

Here we follow the analysis in [78] to find $\mathcal{S}_0$. From (4.22), we see that in order to guarantee $z_i \geq 0$, we need to have $\sqrt{\lambda_0} \leq \sqrt{a_i/(b_i\xi_i)}$, which indicates $f(i) < 1$ for some is. Then, the problem can be stated as: given $\beta_1 \leq \beta_2 \leq \cdots \leq \beta_{\mathrm{N}}$, $f(i_{\mathcal{S}}) < 1$ and $f(i_{\mathcal{S}}+1) \geq 1$, we have

1. $f(i)$ is an increasing function of i for $i \leq i_{\mathcal{S}}$;
2. $f(i) \geq 1$ for $i > i_{\mathcal{S}}$.

Proof. It is straightforward to show that $f(1) < 1$. This indicates that $\mathcal{S}_0 \neq \emptyset$ and thus there exist feasible solutions for $\boldsymbol{z}$. When $i > 1$, we have

$$\begin{aligned} f(i+1) &= \frac{\beta_{i+1}\sum_{j=1}^{i}\sqrt{a_j b_j \xi_j} + b_{i+1}\xi_{i+1}}{\sum_{j=1}^{i} b_j\xi_j + \mathcal{P}_{\mathrm{tot}} + b_{i+1}\xi_{i+1}} \\ &\geq \frac{\beta_{i}\sum_{j=1}^{i}\sqrt{a_j b_j \xi_j} + b_{i+1}\xi_{i+1}}{\sum_{j=1}^{i} b_j\xi_j + \mathcal{P}_{\mathrm{tot}} + b_{i+1}\xi_{i+1}} \\ &\overset{(a)}{\geq} \begin{cases} f(i), & i < i_{\mathcal{S}} \\ 1, & i > i_{\mathcal{S}}. \end{cases} \end{aligned}$$

The first inequality in (a) is valid since when $x/y < 1$, we have $(x+c)/(y+c) \geq x/y$, where $x, y, c > 0$. Then, we see that $f(i)$ is an increasing function of i for $i \leq i_{\mathcal{S}}$. The second inequality in (a) is valid since when $x/y \geq 1$, we have $(x+c)/(y+c) \geq 1$. This indicates that when $f(i) \geq 1$, $f(i+1) \geq 1$ for $i > i_{\mathcal{S}}$. This completes the proof. □

C.4 Solution for Set $\mathcal{S}_1$

Similar to the solution for $\mathcal{S}_0$, we need to show that: given $\tilde{\beta}_1 \leq \tilde{\beta}_2 \leq \cdots \leq \tilde{\beta}_{\mathrm{N}}$, $\tilde{f}(i_{\mathcal{S}}) < 1$ and $\tilde{f}(i_{\mathcal{S}}+1) \geq 1$, we have

1. $\tilde{f}(i) < 1$ for $i \leq i_{\mathcal{S}}$;
2. $\tilde{f}(i) \geq 1$ for $i_{\mathcal{S}} < i \leq \left\lfloor \frac{\mathcal{P}_{\mathrm{tot}}}{\mathcal{P}_{\max}} \right\rfloor$.

Proof. To prove part 1), here we consider 4 cases which depend on the values of $\tilde{\beta}_i$ and β_i: *a)* $\tilde{\mathcal{S}}_i = \tilde{\mathcal{S}}_{i-1} \cup \tilde{\mathcal{S}}'_i \setminus \{i\}$, *b)* $\tilde{\mathcal{S}}_i = \tilde{\mathcal{S}}_{i-1} \cup \tilde{\mathcal{S}}'_i$, *c)* $\tilde{\mathcal{S}}_i = \tilde{\mathcal{S}}_{i-1} \setminus \{i\}$, *d)* $\tilde{\mathcal{S}}_i = \tilde{\mathcal{S}}_{i-1}$, where $\tilde{\mathcal{S}}'_i = \{m | \tilde{\beta}_{i-1} < \beta_m < \tilde{\beta}_i,\ i < m \leq \mathrm{N}\}$. Now we start with case *a*). In case *a*), we have $\beta_i < \tilde{\beta}_{i-1}$ and $\tilde{\mathcal{S}}'_i \neq \emptyset$. Furthermore, we note that

$$\begin{aligned} & \tilde{\beta}_{i-1} \sum_{m \in \tilde{\mathcal{S}}_{i-1}} \sqrt{a_m b_m \xi_m} \\ \leq \ & \tilde{\beta}_i \Big(\sum_{m \in \tilde{\mathcal{S}}_i} \sqrt{a_m b_m \xi_m} + \sqrt{a_i b_i \xi_i} - \sum_{m \in \tilde{\mathcal{S}}'_i} \sqrt{a_m b_m \xi_m} \Big) \\ \leq \ & \tilde{\beta}_i \sum_{m \in \tilde{\mathcal{S}}_i} \sqrt{a_m b_m \xi_m} + (\mathcal{P}_{\max} + b_i \xi_i) - \sum_{m \in \tilde{\mathcal{S}}'_i} b_m \xi_m . \end{aligned}$$

The last inequality is valid since when $m \in \tilde{\mathcal{S}}'_i$, $\beta_m < \tilde{\beta}_i$, which implies

$$\tilde{\beta}_i \sum_{m \in \tilde{\mathcal{S}}'_i} \sqrt{a_m b_m \xi_m} \;\geq\; \sum_{m \in \tilde{\mathcal{S}}'_i} \beta_m \sqrt{a_m b_m \xi_m} \;=\; \sum_{m \in \tilde{\mathcal{S}}'_i} b_m \xi_m.$$

After some manipulations, when $\tilde{f}(i) < 1, \ \forall i < i_{\mathcal{S}}$, we have

$$\tilde{f}(i-1) \;\leq\; \frac{\tilde{\beta}_i \sum_{m \in \tilde{\mathcal{S}}_i} \sqrt{a_m b_m \xi_m} + c_1}{(\mathcal{P}_{\text{tot}} - i\mathcal{P}_{\max} + \sum_{m \in \tilde{\mathcal{S}}_i} b_m \xi_m) + c_1} \;<\; 1,$$

where $c_1 = (\mathcal{P}_{\max} + b_i \xi_i) - \sum_{m \in \tilde{\mathcal{S}}'_i} b_m \xi_m$. The last inequality is valid because when $x/y < 1$, we have $(x + c_1)/(y + c_1) < 1$, where $x, y > 0$ and $c_1 > -x$. Similarly, we see that for other three cases, we also have $\tilde{f}(i-1) < 1$.

Now let us prove part 2). Similarly, for case a), we have

$$\tilde{f}(i+1) \;\geq\; \frac{\tilde{\beta}_i \sum_{m \in \tilde{\mathcal{S}}_i} \sqrt{a_m b_m \xi_m} - c_2}{(\mathcal{P}_{\text{tot}} - i\mathcal{P}_{\max} + \sum_{m \in \tilde{\mathcal{S}}_i} b_m \xi_m) - c_2} \;\geq\; 1.$$

where $c_2 = (\mathcal{P}_{\max} + b_{i+1} \xi_{i+1}) - \sum_{m \in \tilde{\mathcal{S}}'_{i+1}} b_m \xi_m$. The last inequality is valid because when $x/y \geq 1$, we have $(x - c_2)/(y - c_2) \geq 1$, where $x, y > 0$ and $c_2 < y$. Similarly, we see that for other three cases, we have $\tilde{f}(i+1) \geq 1$. This completes the proof. □

Vita

Gang Xiong was born in 1976 in Hubei Province, P. R. China. He received his B.S. and M.S. degree in precision instrument and control from Harbin Institute of Technology, Harbin, P. R. China in 1997 and 1999, respectively. He then received his M.S. degree in electrical engineering from the Ohio State University, Columbus, OH in 2003, and is currently working toward his Ph.D. degree in the department of electrical and computer engineering at Lehigh University, Bethlehem, PA.

From 2003 to 2007, he was a team leader at DSP Department, ZTE Corporation, Shanghai, P. R. China, where he led a baseband group for baseband design and DSP development of GSM/EDGE base station system, and he was involved in baseband design of TD-SCDMA and WiMAX systems. His current research interests include PHY and MAC layer wireless communication, cross-layer design and optimization of wireless networks.

LIST OF PUBLICATIONS

JOURNAL PAPERS

1. G. Xiong and S. Kishore, “Cooperative Spectrum Sensing with Beamforming in Cognitive Radio Networks,” *IEEE Commun. Lett.,* vol. 15, no. 2, pp. 220-222, Feb. 2011.

2. G. Xiong and S. Kishore, “Linear High-Order Distributed Average Consensus Algorithm in Wireless Sensor Networks,” *EURASIP Journal on Advances in Signal Processing*, vol. 2010, Article ID 373604.

3. G. Xiong and S. Kishore, “Discrete-Time Second-Order Distributed Consensus Time Synchronization Algorithm for Wireless Sensor Networks,” *EURASIP Journal on Wireless Communications and Networking*, vol. 2009, Article ID 623537.

4. G. Xiong and S. Kishore, “Analysis of Distributed Consensus Time Synchronization with Gaussian Delay in Wireless Sensor Networks,” *EURASIP Journal on Wireless Communications and Networking*, vol. 2009, Article ID 528161.

5. G. Xiong, S. Kishore and A. Yener, “Spectrum Sensing in Cognitive Radio Networks: Performance Evaluation and Optimization,” *In preparation for journal submission.*

6. G. Xiong and S. Kishore, “Joint Transmitter and Receiver Design with Adaptive Beamforming in MIMO SC-FDMA Systems,” *Submitted to IEEE Trans. Wireless Commun..*

CONFERENCE PAPERS

1. G. Xiong and S. Kishore, “Joint Transmitter and Receiver Design with Adaptive Beamforming in MIMO SC-FDMA Systems,” *Accepted in International Conference on Acoustics, Speech and Signal Processing (ICASSP) 2011.*

2. G. Xiong, C. Chen, S. Kishore and A. Yener, “Smart (In-home) Power Scheduling for Demand Response on the Smart Grid,” *2011 PES Innovative Smart Grid Technologies Conference*, Anaheim, CA, Jan. 2011.

3. G. Xiong, S. Kishore and A. Yener, “On Performance Evaluation of Cooperative Spectrum Sensing in Cognitive Radio Networks,” *Conference on Information Sciences and Systems (CISS)*, Princeton, NJ, pp. 1-6, March 2010.

4. G. Xiong, S. Kishore and A. Yener, “Cost Constrained Spectrum Sensing in Cognitive Radio Networks,” *Conference on Information Sciences and Systems (CISS)*, Princeton, NJ, pp. 1-6, March 2010.

5. G. Xiong, S. Kishore and A. Yener, “On Low Complexity Cooperative Spectrum Sensing for Cognitive Networks,” (*Invited Paper*), *Third International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP09)*, Aruba, Dutch Antilles, Dec., 2009.

6. G. Xiong and S. Kishore, “Consensus-Based Distributed Detection Algorithm in Wireless Ad Hoc Networks,” *3rd International Conference on Signal Processing and Communication Systems (ICSPCS)*, Omaha, NE, Sept. 2009.

7. G. Xiong and S. Kishore, “Linear High-Order Distributed Average Consensus Algorithm in Wireless Sensor Networks,” *IEEE Workshop on Statistical*

Signal Processing (SSP09), Cardiff, Wales, Aug. 2009.

8. G. Xiong and S. Kishore, "Performance of Distributed Consensus Time Synchronization with Gaussian Delay in Wireless Sensor Networks," *Wireless Communications and Networking Conference (WCNC)*, Budapest, Hungary, pp. 1-5, April 2009.

9. G. Xiong and S. Kishore, "Cooperative Spectrum Sensing in Cognitive Radio Networks Under Gaussian Fusion Channels," *Conference on Information Sciences and Systems (CISS)*, Baltimore, MD, pp. 675-680, March 2009.

10. G. Xiong and S. Kishore, "Second Order Distributed Consensus Time Synchronization Algorithm for Wireless Sensor Networks," *IEEE Global Telecommunications Conference (GLOBECOM)*, New Orleans, LA, pp. 1-5, Nov. 2008.

11. G. Xiong, O. Y. Takeshita, M. P. Fitz and W. Zhu. "Design and Implementation of a Preamble-based Burst Mode CPM Modem over Rayleigh Fading Channels." *Proc. SPIE*, Vol. 5205, San Diego, CA, 2003, pp. 242-253.