OPTIMAL RESOURCE ALLOCATION AND CROSS-LAYER CONTROL IN COGNITIVE AND COOPERATIVE WIRELESS NETWORKS

by

Rahul Urgaonkar

A Dissertation Presented to the
FACULTY OF THE USC GRADUATE SCHOOL
UNIVERSITY OF SOUTHERN CALIFORNIA
In Partial Fulfillment of the
Requirements for the Degree
DOCTOR OF PHILOSOPHY
(ELECTRICAL ENGINEERING)

May 2011

Dedication

To Aai-Baba, BB, and Paulami

Acknowledgments

This work would not have been possible without the encouragement, help, and guidance that I received over the years from many individuals. First and foremost is my advisor, Prof. Michael J. Neely. I remember attending a seminar talk that he gave at USC as a new faculty member sometime in early 2004. At that time, I was a Master's student, barely contemplating the idea of pursuing a Ph.D. I still remember the sense of awe that I felt listening to him talk about his doctoral work. I thought that if I were to ever pursue a Ph.D., this is the kind of research I would want to do. I am forever grateful to Prof. Neely for taking me as his student and patiently helping me through thick and thin, from being a role model and a guide to sharing his candid assessments of my work and providing valuable feedback. I hope I have been able to achieve at least some of what I set out to do in my work.

I would like to express my sincere thanks to Prof. Bhaskar Krishnamachari who was my research advisor during my Master's studies and who served on my committee. It was with Bhaskar that I got the first opportunity to do research in the area of Wireless Networks. His enthusiasm for research, learning, and mentoring students is truly inspiring and I feel lucky to have benefitted from that. I am also thankful to my other committee members: Prof. Giuseppe Caire, Prof. Leana Golubchik, Prof. Keith M. Chugg, and Prof. C.-C. Jay Kuo for their time.

Working in the Communication Sciences Institute was made memorable by my friends and colleagues Ozgun, Chih-ping, and Longbo as well as Gerrie, Anita, Mayumi, and Milly. I also thank Apoorva, my room-mate for several years, and Sumit and Anant, for their friendship. I wish everyone the best in all future endeavors of their lives.

My family has been the constant force supporting me religiously in my darkest hours. I could count on them to discuss with me at any hour what now seem the most trivial of all "problems". My parents, Narendra and Dr. Charu Urgaonkar, have been my greatest fans and this work is dedicated to them. My brother, Prof. Bhuvan Urgaonkar has been a role model and a friend whose advice has helped me in innumerable ways. I am also thankful to my Uncle and Aunt, Dr. Mohan and Shachi Gawande and my grandfather, Dr. Trimbak Gawande for providing me a home away from home.

Last, but not the least of all who helped me jump "The Fence" is my best friend, my dear wife Paulami whose love for me knows no bounds. The best thing that ever happened to me was meeting with her. Thank you Paulami, for making every day with you an adventure!

Rahul Urgaonkar

March 2011

Table of Contents

List of Figures

Abstract

We investigate four problems on optimal resource allocation and cross-layer control in cognitive and cooperative wireless networks with time-varying channels. The first three problems consider different models and capabilities associated with cognition and cooperation in such networks. Specifically, the first problem focuses on the *dynamic spectrum access* model for cognitive radio networks and assumes no cooperation between the licensed (or "primary") and unlicensed (or "secondary") users. Here, the secondary users try to avoid interfering with the primary users while seeking transmission opportunities on vacant primary channels in frequency, time, or space. The second problem considers a *relay-based fully cooperative* wireless network. Here, cooperative communication techniques at the physical layer are used to improve the reliability and energy cost of data transmissions. The third problem considers a *cooperative cognitive radio network* where the secondary users can cooperatively transmit with the primary users to improve the latter's effective transmission rate. In return, the secondary users get more opportunities for transmitting their own data when the primary users are idle.

In all of these scenarios, our goal is to design optimal control algorithms that maximize time-average network utilities (such as throughput) subject to time-average constraints (such as power, reliability, etc.). To this end, we make use of the technique of Lyapunov

optimization to design online control algorithms that can operate without requiring any knowledge of the statistical description of network dynamics (such as fading channels, node mobility, and random packet arrivals) and are provably optimal. The algorithms for the first two problems use greedy decisions over one slot and two-slot frames, whereas the algorithm for the third problem involves a stochastic shortest path decision over a variable length frame, and this is explicitly solved, remarkably without requiring knowledge of the network arrival rates.

Finally, in the fourth problem, we investigate optimal routing and scheduling in static wireless networks with *rateless codes*. Rateless codes allow each node of the network to accumulate mutual information with every packet transmission. This enables a significant performance gain over conventional shortest path routing. Further, it also outperforms cooperative communication techniques that are based on energy accumulation. However, it requires complex and combinatorial networking decisions concerning which nodes participate in transmission, and which decode ordering to use. We formulate the general problems as combinatorial optimization problems and identify several structural properties of the optimal solutions. This enables us to derive optimal greedy algorithms to solve these problems. This work uses a different set of tools and can be read independently of the other chapters.

Chapter 1

Introduction

Next generation wireless networks are expected to provide significantly higher data rates, reliability, and energy efficiency than the current systems. There has been much effort in recent years to develop new techniques that improve the performance of wireless networks to achieve these objectives. Cognitive radio and cooperative communication are two important examples of such emerging techniques.

The motivation for cognitive radios comes from the observation that the existing static allocation of spectrum to licensed (or "primary") users leads to inefficient utilization and creates spectrum scarcity. By allowing unlicensed (or "secondary") wireless devices to dynamically access the unused portions of the spectrum, it is possible to support more users in the existing spectrum and improve its spectral efficiency. However, such dynamic spectrum access may cause undesirable interference to the licensed users. Thus, it is important to design opportunistic scheduling schemes that provide strong reliability guarantees for the licensed users while attempting to maximize the utility (e.g., throughput) of the unlicensed users.

The motivation for cooperative communication comes from the work on MIMO systems which shows that deploying multiple antennas on wireless devices offers substantial performance improvements. However, this may be infeasible is small-sized devices due to space limitations. Cooperative communication ("network MIMO") tries to emulate the gains of traditional MIMO systems in a distributed network of single antenna nodes. This form of communication transforms the traditional node or link based problems of resource allocation into a network wide problem. This necessitates the design of opportunistic algorithms that make use resources (such as power) fairly across all users to achieve a target performance.

The technique of cooperative communication can be used to obtain further gains in cognitive radio networks that go beyond the traditional *dynamic spectrum access* model. In this model, the secondary users try to avoid interfering with the primary users while seeking transmission opportunities on vacant primary channels. This model assumes no cooperation between the primary and secondary users. However, with cooperative communication, a secondary user can use its resources to improve the effective transmission rate of the primary user. In return, the secondary user can get more opportunities for transmitting its own data when the primary user is idle. In this scenario, the secondary users need to make dynamic decisions on whether to cooperate or not in order to maximize their transmission opportunities.

In this thesis, we study several such resource allocation problems in the area of cognition and cooperation in wireless networks. Our goal is to design optimal control algorithms that maximize general time-average network utilities (such as throughput) subject to time-average constraints (such as power, reliability, etc.). We describe these problems in more detail in Sec. 1.2.

1.1 Models for Cognitive Radio Networks

Several different models for cognitive radio networks have been considered in the literature. Depending on the assumptions made about the capabilities of cognition and cooperation and the method of secondary user transmissions, these can be broadly classified under the following three categories [GJMS09]:

1. **Underlay Model**: In this model, the secondary users are allowed to transmit concurrently with the primary users as long as the resulting interference caused to the primary receivers is below some acceptable threshold. This may be achieved, for example, using ultrawideband (UWB) transmissions where the secondary users transmit over a wide bandwidth such that the resulting interference power at the primary receivers is below the noise floor. Since the primary interference constraints are typically quite restrictive, the secondary users are limited to short range communication in this model.

2. **Overlay Model**: In this model, it is assumed that the secondary users are aware of the primary user codebooks and possibly its messages. This knowledge can then be exploited by the secondary user to either mitigate or altogether cancel any interference seen at the primary and secondary receivers. This may be achieved using sophisticated coding and interference management techniques such as Dirty Paper Coding and Interference Alignment. While this model can potentially achieve

the largest rate region, the assumption about non-causal knowledge of primary messages at the secondary user may limit its practical utility.

3. **Interweave Model:** This model is inspired by the notion of *opportunistic communication* where the secondary users seek transmission opportunities in vacant primary channels in frequency, time, and/or space, also knows as "spectrum holes". Also referred to as the *dynamic spectrum access* model, here the secondary users monitor the spectrum occupancy process of the primary users and then opportunistically transmit on idle primary channels. A key challenge here is to maximize such opportunities while limiting the interference caused to the primary users due to imperfect knowledge of the primary user channel occupancy state.

In this thesis, we will focus primarily on the **Interweave Model**. Within this model, several variants have been considered in the literature that differ in the assumptions made on the interaction between the primary and secondary users in the network. See, for example, [Bud07, ZS07] for surveys on the taxonomy and classifications for such dynamic spectrum access networks. On one extreme is the case where the primary users are completely oblivious to the secondary users and do not change their spectrum usage to accommodate them. In this case, it is the responsibility of the secondary users to avoid interfering excessively with the primary users by intelligently monitoring the spectrum and transmitting opportunistically. On the other extreme is the case where the primary and secondary users fully cooperate in each other's transmissions (for example, using relay-based cooperative communication). There can also be hybrid scenarios where the primary users are aware of the presence of secondary users, but do not spend their resources helping secondary transmissions. We consider all of these scenarios in Chapters 2, 3, and 4 respectively, as discussed next.

1.2 Summary of Contributions

In this thesis, we study the following problems on optimal resource allocation and cross-layer control in cognitive and cooperative wireless networks:

1. In Chapter 2, we consider a cognitive network with licensed (primary) users and unlicensed (secondary) users under the dynamic spectrum access model. The primary users are assumed to be completely oblivious to the presence of the secondary users. The secondary users have imperfect knowledge about the primary users' spectrum usage and must meet a constraint on the maximum time-average rate of collisions for each primary user while seeking transmission opportunities on the primary channels. We formulate this as a constrained stochastic optimization problem. In order to satisfy the maximum collision constraint, we make use of the virtual cost queue technique of [Nee06] in the form of "collision" queues. These collision queues enable stochastic optimization by acting as dynamic Lagrange multipliers [HN09]. Using the technique of Lyapunov optimization, we design an online admission control, scheduling and resource allocation algorithm that meets the desired objectives and provides explicit performance guarantees. This algorithm works in the presence of imperfect knowledge about the primary user spectrum usage and does not require

knowledge of the secondary user mobility patterns. A salient feature of our algorithm is that it provides *deterministic* worst case bounds on the maximum number of collisions suffered by a primary user over any time duration.

2. In Chapter 3, we investigate optimal resource allocation for *delay-limited* cooperative communication in time varying wireless networks. Specifically, we consider a team of mobile users with real-time applications that have strict delay constraints and fixed rate and reliability requirements (e.g., voice, multimedia). Cooperative communication is particularly attractive in such delay-limited scenarios since it can offer significant spatial diversity gains on top of conventional techniques used for combating fading. In this chapter, we develop dynamic cooperation strategies that make optimal use of network resources to achieve a target outage probability (reliability) for each user subject to average power constraints. Using the technique of Lyapunov optimization, we first present a general framework to solve this problem and then derive quasi-closed form solutions for several cooperative protocols proposed in the literature (such as Decode-and-Forward and Amplify-and-Forward). Both scenarios where channel state information is available at the transmitter and when only the statistics are known are considered. The model studied in this chapter can be considered as a fully cooperative cognitive network where there is no distinction between the primary and secondary users.

3. In Chapter 4, we extend the model of a cognitive radio network introduced in Chapter 2 and allow a secondary user to cooperate with the primary user in order to improve the reliability of the primary transmissions. Although the secondary user

must use its own resources for such cooperation, the observation is that this potentially creates more opportunities for the secondary user to transmit its data. However, the secondary user must balance the desire to cooperate more (to create more transmission opportunities) with the need for maintaining sufficient energy levels for its own transmissions. Such a model is applicable in the emerging area of cognitive femtocell networks. We formulate the problem of maximizing the secondary user throughput subject to a time average power constraint under these settings. This is a constrained Markov Decision Problem and conventional solution techniques based on dynamic programming require either extensive knowledge of the system dynamics or learning based approaches that suffer from large convergence times. However, using the technique of Lyapunov optimization, we design a novel *greedy* and *online* control algorithm that overcomes these challenges and is provably optimal.

4. In Chapter 5, we consider the problem of optimal routing and scheduling strategies for multi-hop wireless networks with *rateless codes*. Rateless codes allow each node of the network to accumulate mutual information with every packet transmission. This enables a significant performance gain over conventional shortest path routing. Further, it also outperforms cooperative communication techniques that are based on energy accumulation. However, it requires complex and combinatorial networking decisions concerning which nodes participate in transmission, and which decode ordering to use. We formulate the general problem as a combinatorial optimization problem and then make use of several structural properties to simplify the solution

and derive an optimal greedy algorithm. Although the reduced problem still has exponential complexity, using the insight obtained from the optimal solution to a line network, we propose two simple heuristics that can be implemented in polynomial time in a distributed fashion and compare them with the optimal solution. Simulations suggest that both heuristics perform very close to the optimal solution over random network topologies.

1.3 Outline of Thesis

Cognitive radio networks and cooperative communication are expected to be essential components of future wireless networks. The research performed in this thesis investigates optimal resource allocation and network control problems in these areas using deterministic and stochastic optimization techniques. Specifically, the analysis presented in Chapters 2, 3, and 4 is based on the framework of cross-layer design using Lyapunov optimization theory [GNT06,Nee10b]. Control algorithms developed using this stochastic optimization approach have several attractive features. In particular, they do not require knowledge of the statistics of the packet arrival, user mobility and channel fading processes. These algorithms are *greedy* and *online* and thus amenable to implementation. Chapter 5, which considers deterministic and combinatorial optimization problems, uses a different set of analytical tools and can be read independently of the other chapters.

Chapter 2

Reliable Scheduling in Cognitive Radio Networks

This chapter focuses on reliable scheduling in cognitive radio networks consisting of both primary (licensed) and secondary (unlicensed) users. Specifically, we consider the *dynamic spectrum access* model for cognitive radio networks in which the secondary users seek transmission opportunities on vacant primary channels in frequency, time, or space. However, the current primary channel occupancy state is not fully known to the secondary users. Rather, we assume that they only know the probability of a primary channel being busy at any given time. In this setting, we formulate the problem of maximizing the sum total throughput utility of the secondary users subject to time-average collision constraints with the primary users. Using the technique of Lyapunov optimization, we construct an online control algorithm that jointly performs admission control, scheduling and resource allocation and provides explicit performance guarantees. A key feature of this algorithm is its use of "collision" queues that enable it to provide tight reliability guarantees in the form of a bound on the worst case number of collisions suffered by a primary user in any time interval. This algorithm operates without requiring a-priori knowledge of the mobility patterns of the secondary users and yields an average throughput utility that can be pushed arbitrarily close to the optimal value, with a trade-off in average delay.

2.1 Introduction

Cognitive radio networks have recently emerged as a promising technique to improve the utilization of the existing radio spectrum. The key enabler is the cognitive radio [Mit00, MM99, Hay05] that can dynamically adjust its operating points over a wide range depending on spectrum availability. The main idea behind a cognitive network is for the unlicensed users to exploit the spatially and/or temporally underutilized spectrum by transmitting *opportunistically*. However, a basic requirement is to ensure that the existing licensed users are not adversely affected by such transmissions. Such interference with the licensed users may be unavoidable due to lack of precise channel state information. In this chapter, we develop an opportunistic scheduling algorithm that maximizes the throughput utility of the secondary (or unlicensed) users subject to maximum collision constraints with the primary (or licensed) users in a cognitive radio network. This algorithm works in the presence of imperfect knowledge about primary user spectrum usage and provides tight reliability guarantees.

A survey on the taxonomy, design issues, and recent work in cognitive radio networks is provided in [ALVM06, Bud07, ZS07]. The problem of optimal spectrum assignment to secondary users in static networks is treated in [PZZ06, CZ05, WLX05, HSS07, YBC+07, SH08, DSM09] where it is assumed that scheduling is aware of primary user transmissions.

Scheduling the secondary users under partial channel state information is considered in [CZS08, ZTSC07, HLD08, LKL10] which use a probabilistic maximum collision constraint with the primary users.

In this chapter, we use the techniques of adaptive queueing and Lyapunov optimization to design an online admission control, scheduling and resource allocation algorithm for a cognitive network that maximizes the throughput utility of the secondary users subject to a maximum rate of collisions with the primary users. This algorithm operates without knowing the mobility pattern of the secondary users and provides explicit performance bounds. Lyapunov optimization techniques were perhaps first applied to wireless networks in the landmark paper [TE92], where Lyapunov drift is used to develop a joint optimal routing and scheduling algorithm. This method has since been extended to treat problems of joint stability and utility optimization in general stochastic networks in [Nee03,NMR05, NML08, Nee06] and wireless mesh networks in [NU07]. Recent work in [KS10, LLS10] applies these techniques for resource allocation problems in cognitive radio networks, similar to our work in this chapter. The analysis presented in all of these works, including this chapter is based on the framework of Lyapunov optimization theory [GNT06,Nee10b].

The main contributions of this chapter are described below:

- We develop throughput optimal control policies for cognitive networks with general interference and mobility models.

- We introduce the notion of "collision" queues that are used to provide strong *reliability* bounds in terms of the worst case number of collisions suffered by a primary user in any time interval. In particular, the collision queue method here is adapted from the virtual power queue technique of [Nee06]. However, the collision queues developed here are designed to ensure reliability constraints, rather than average power constraints. Different from [Nee06], this requires the inputs to the virtual queues to be random collision variables that can be evaluated only after packet transmission has taken place.

- We develop easier to implement constant factor approximations to the optimal resource allocation problem.

The rest of the chapter is organized as follows. We describe the network model and assumptions in Sec. 2.2. We formulate the objective of maximizing the sum throughput utility of the secondary users subject to time average collision constraints as a stochastic optimization problem in Sec. 2.3. Then, in Sec. 2.4, we present an online control algorithm CNC that solves this problem optimally. Subsequent sections analyze its performance and provide analytical guarantees. In Sec. 2.6, we describe a distributed version of CNC and provide simulation based evaluation in Sec. 2.7.

2.2 Network Model

We consider a cognitive radio network consisting of M primary users and N secondary users as shown in Fig. 2.1. Each primary user has a unique licensed channel and these are orthogonal in frequency and/or space. Thus, the primary users can send data over their own licensed channels to their respective access points simultaneously. The secondary users do not have any such channels and opportunistically try to send their data to their

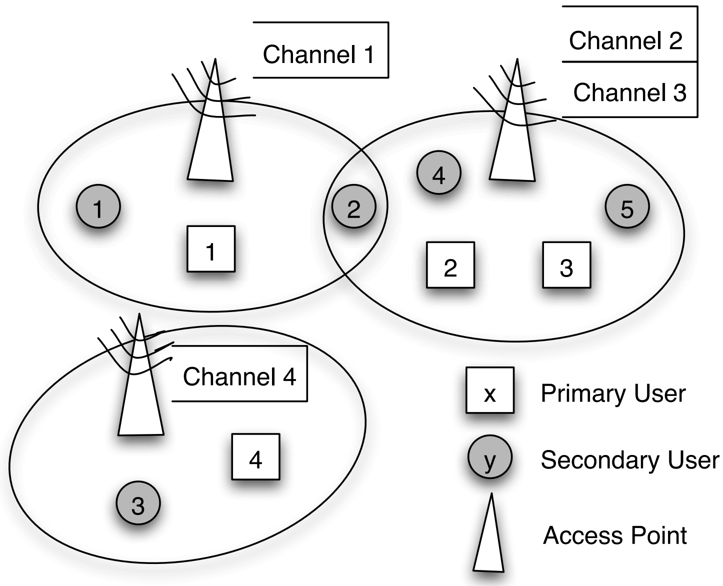

Figure 2.1: Example cognitive network showing primary and secondary users

receivers by utilizing idle primary channels. Such opportunities are called "spectrum holes" [TSM09].

2.2.1 Mobility Model

We consider a time-slotted model. The primary users are assumed to be static. However, the secondary users could be mobile so that the set of channels they can access can change over time. In a timeslot, a secondary user can access a subset of the primary channels potentially depending on its current location. This information is concisely represented by an $N \times M$ binary channel accessibility matrix $\boldsymbol{H}(t) = \{h_{nm}(t)\}_{N\times M}$ where:

$$h_{nm}(t) = \begin{cases} 1 & \text{if secondary user } n \text{ can access channel } m \text{ in slot } t \\ 0 & \text{else} \end{cases}$$

For example, the channel accessibility matrix for the example network in Fig.2.1 is given by:

$$\boldsymbol{H}(t) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

Specifically, secondary user 1 in Fig. 2.1 can currently access channel 1 only (as indicated by the first row of the $\boldsymbol{H}(t)$ matrix above), while secondary user 2 can currently access either channels $1, 2$, or 3 (as indicated by the second row in the $\boldsymbol{H}(t)$ matrix). We assume that the mobility process of the secondary users is such that the resulting $\boldsymbol{H}(t)$ process is Markovian and has a well defined steady state distribution. However, the transition probabilities associated with this Markov Chain could be unknown.

2.2.2 Interference Model

Let $\boldsymbol{S}(t) = (S_1(t), S_2(t), \ldots, S_M(t))$ represent the current primary user occupancy state of the M channels. Here, $S_i(t) \in \{0, 1\}$ (for $i \in \{1, 2, \ldots, M\}$) with the interpretation that $S_i(t) = 0$ if channel i is occupied by primary user i in timeslot t and $S_i(t) = 1$ if primary user i is idle in timeslot t. We assume that exactly 1 packet can be transmitted over any channel in a timeslot. A secondary user can attempt transmission over at most 1 channel subject to the constraints in $\boldsymbol{H}(t)$. This transmission is successful only when the channel is not being used by its primary user or any other secondary user. If a secondary user transmits on a channel which is busy, there is a collision and both packets are lost. We assume that multi-user detection/interference cancellation is not available so that if the secondary user attempts to transmit its own data when some other user is also transmitting, there is enough interference at the access point and no data is successfully received.

To capture the interference that a secondary user transmission may cause on other channels, for all $n \in \{1, 2, \ldots, N\}, m \in \{1, 2, \ldots, M\}$, we define $\mathcal{I}_{nm}(t)$ as the set of channels that secondary user n interferes with when it uses channel m in timeslot t . We include m in the set $\mathcal{I}_{nm}(t)$. We further define the following indicator variables (to be used later):

$$I^k_{nm}(t) = \begin{cases} 1 & \text{if } k \in \mathcal{I}_{nm}(t) \qquad \forall\, k \in \{1, 2, \ldots, M\} \\ 0 & \text{else} \end{cases}$$

Clearly, $I^m_{nm}(t) = 1$ for all m, n, t. Under this interference model, the following two conditions are necessary for a transmission by secondary user n on channel m in slot t to be successful:

1. $S_m(t) = 1$

2. For all other secondary users i transmitting on a channel $j \in \{1, 2, \ldots, M\}$, we have $m \notin \bigcup \mathcal{I}_{ij}(t)$ (where $i \in \{1, 2, \ldots, N\} \setminus \{n\}$)

This interference model is general enough to capture scenarios in which the channels may not be orthogonal with respect to the secondary user transmissions although they are orthogonal for the primary user transmissions. Further, it is general enough to model

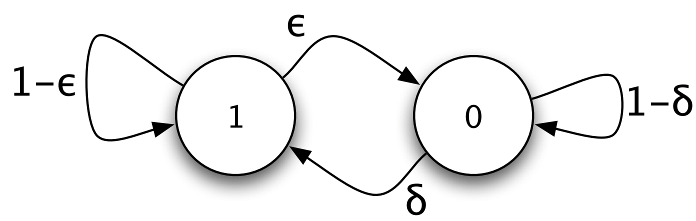

Figure 2.2: Two state Markov Chain example for primary user channel occupancy process

scenarios where these sets could also change over time (possibly depending on the secondary user location). In most practical situations, the cardinality of the interference sets $\mathcal{I}_{nm}(t)$ would be small. An important special case is when the channels are indeed orthogonal for all secondary user transmissions, so that $\mathcal{I}_{nm}(t) = \{m\}$ for all m, n, t.

As an example, consider secondary user 4 in Fig. 2.1, and suppose this user transmits a packet over channel 2. Under an orthogonal channel model, we would have $\mathcal{I}_{42}(t) = \{2\}$, as this transmission would not interfere with any other channels. However, in a model where channels are not necessarily orthogonal, it might be that channel 2 uses the same frequency as channel 1, in which case we would have $\mathcal{I}_{42}(t) = \{2, 1\}$, as the current location of node 4 may be close enough to interfere with channel 1 (even though it is not close enough to communicate over channel 1). Note that this $\mathcal{I}_{42}(t)$ set can potentially change over time if node 4 moves to a location that would no longer would interfere with channel 1.

2.2.3 Primary User Traffic Model

We assume that the primary user channel occupancy process $\boldsymbol{S}(t)$ evolves according to a finite state ergodic Markov Chain on the state space $\{0, 1\}^M$ and is independent of the secondary user mobility process $\boldsymbol{H}(t)$. It is further assumed to be independent of the control actions of the secondary users. In particular, we assume that the primary users do not attempt retransmissions when collisions take place. For example, the primary users may be using a voice application which can tolerate some lost packets, but has strict delay constraints so that retransmissions are not done. Another example is where the primary users use erasure codes such that the data can be recovered even when some packets are lost.

Each primary user m receives exogenous data at a rate $\nu_m \leq 1$ packet/slot and can tolerate a maximum time average rate of collisions given by $\rho_m \nu_m$, where $\rho_m < 1$ is the maximum fraction of primary user m packets that can have collisions and is known to the secondary users. For example, $\rho_m = 0.05$ means that at most 5% of primary user m packets can have collisions.

2.2.4 Channel State Information Model

The channel state information available to the secondary users is described by a probability vector $\boldsymbol{P}(t) = (P_1(t), P_2(t), \ldots, P_M(t))$ where $P_i(t)$ is the probability that primary user i is idle in timeslot t. The $\boldsymbol{P}(t)$ process is assumed to be *modulated* by a finite state discrete time Markov Chain (DTMC). Specifically, let $\chi(t)$ represent a finite state DTMC that represents the state of the primary users (where "state" is an abstract term here and could be different in different examples, e.g., it could be $\boldsymbol{S}(t-1)$, the channel occupancy state in the previous slot). The $\chi(t)$ process is assumed to be independent of the control actions. Then for each channel m and each slot t, we define $P_m(t) = Pr[S_m(t) = 1 | \chi(t)]$.

Thus, $P_m(t)$ is modulated by this process and hence is also independent of the control actions.

We assume that this information is obtained either through a knowledge of the traffic statistics of the primary users, or by sensing the channels, or a combination of these. In addition, prediction based techniques could also be used to get this information. We discuss two examples of these scenarios in the following.

Example 1: Using knowledge of traffic statistics: Consider a single primary user whose channel occupancy process $S(t)$ is described by a 2 state Markov Chain as shown in Fig. 2.2. Suppose the last state of the Markov Chain is known at the beginning of each slot and let $\chi(t) = S(t-1)$. If the transition probabilities ϵ and δ associated with this Markov Chain are known, then one can compute $P(t) = Pr[S(t) = 1|S(t-1)]$. Specifically, $Pr[S(t) = 1|S(t-1) = 0] = \delta$ and $Pr[S(t) = 1|S(t-1) = 1] = 1 - \epsilon$. A secondary user can obtain this information, for example, by querying the primary user base station that knows $\chi(t)$, so that it is able to tell the current $P(t)$ value. It can be seen that in this example $P(t)$ is modulated by the 2 state $\chi(t)$ process.

Example 2: Using a combination of channel sensing and traffic statistics: In the example above, suppose a secondary user also *senses* the current channel state $S(t)$ and uses a detection algorithm that outputs $\tilde{S}(t)$ as follows:

$$\text{if } S(t) = 0,\ \tilde{S}(t) = \begin{cases} 1 & \text{w.p. } p \\ 0 & \text{w.p. } 1-p \end{cases} \qquad \text{if } S(t) = 1,\ \tilde{S}(t) = \begin{cases} 1 & \text{w.p. } 1-q \\ 0 & \text{w.p. } q \end{cases}$$

Here, p and q can be thought of as the probabilities of false detection associated with the sensing mechanism. Similar models have been considered in [CZS08, ZTSC07].

Let $\chi(t) = [\tilde{S}(t), S(t-1)]$. Then, a secondary user can compute $P(t)$ as follows:

If $\tilde{S}(t) = 1$:

$$\begin{aligned}
P(t) &= Pr[S(t) = 1|\tilde{S}(t) = 1, S(t-1)] \\
&= Pr[\tilde{S}(t) = 1|S(t) = 1, S(t-1)]\frac{Pr[S(t) = 1|S(t-1)]}{Pr[\tilde{S}(t) = 1|S(t-1)]} \\
&= \frac{(1-q)Pr[S(t) = 1|S(t-1)]}{(1-q)Pr[S(t) = 1|S(t-1)] + pPr[S(t) = 0|S(t-1)]}
\end{aligned}$$

If $\tilde{S}(t) = 0$:

$$\begin{aligned}
P(t) &= Pr[S(t) = 1|\tilde{S}(t) = 0, S(t-1)] \\
&= Pr[\tilde{S}(t) = 0|S(t) = 1, S(t-1)]\frac{Pr[S(t) = 1|S(t-1)]}{Pr[\tilde{S}(t) = 0|S(t-1)]} \\
&= \frac{qPr[S(t) = 1|S(t-1)]}{qPr[S(t) = 1|S(t-1)] + (1-p)Pr[S(t) = 0|S(t-1)]}
\end{aligned}$$

In this example too, it can be seen that $P(t)$ is modulated by the $\chi(t)$ process.

Our model for the channel state information captures the situations where the exact channel state may not be available to the secondary users (e.g., due to limitations in carrier sensing). These probabilities capture the inherent sensing measurement errors associated with any primary transmission detection algorithm. Intuitively, the "closer" $\boldsymbol{P}(t)$ is to $\boldsymbol{S}(t)$, the smaller the chances of collisions.

2.2.5 Queueing Dynamics and Control Decisions

Each secondary user n receives data according to an arrival process $A_n(t)$ that has rate λ_n packets/slot. We assume that the maximum number of arrivals to any secondary user n is upper bounded by a constant value A_{max} every timeslot. This data arrives at the transport layer and admission control decisions on how many packets to admit to the network layer are taken by each secondary user. We assume that there are no transport layer buffers and add/drop decisions are taken immediately.

Let $Q_n(t)$ be the backlog in the network layer queue of secondary user n at the beginning of timeslot t. Let $R_n(t)$ be the control decision that denotes the number of new packets admitted into this queue in slot t. Define $\mu_{nm}(t)$ as the control decision that allocates channel m to secondary user n in slot t. In this model $\mu_{nm}(t) \in \{0, 1\} \; \forall \; m, n$ with the interpretation that $\mu_{nm}(t) = 1$ if secondary user n transmits on channel m and $\mu_{nm}(t) = 0$ else. Note that there is a successful transmission on channel m only when the necessary conditions specified earlier are met. Then the queueing dynamics of secondary user n under these control decisions is described by:

$$Q_n(t+1) = \max[Q_n(t) - \sum_{m=1}^{M} \mu_{nm}(t) S_m(t), 0] + R_n(t) \tag{2.1}$$

where

$$\mu_{nm}(t) \in \{0, 1\} \; \forall \; m, n \tag{2.2}$$

$$\mu_{nm}(t) \leq h_{nm}(t) \; \forall \; m, n \tag{2.3}$$

$$0 \leq \sum_{m=1}^{M} \mu_{nm}(t) \leq 1 \; \forall \; n \tag{2.4}$$

$$\mu_{nm}(t) = 1 \iff \sum_{j=1}^{M} \sum_{\substack{i=1 \\ i \neq n}}^{N} I_{ij}^{m}(t) \mu_{ij}(t) = 0 \; \forall \; m, n \tag{2.5}$$

$$0 \leq R_n(t) \leq A_n(t) \tag{2.6}$$

Here, inequality (2.3) represents the constraint imposed by the channel accessibility matrix $\boldsymbol{H}(t)$. Inequality (2.4) represents the constraint that a secondary user can be allocated at most 1 channel. (2.5) represents the second necessary condition for successful transmission expressed in terms of the $I_{nm}^{k}(t)$ variables. In the special case of orthogonal channels, this simplifies to the constraint that a channel can be allocated to at most 1 secondary user, i.e.,

$$0 \leq \sum_{n=1}^{N} \mu_{nm}(t) \leq 1 \; \forall \; m \tag{2.7}$$

2.2.6 Discussion of Network Model

The above network model considers access point based networks with static (or locally mobile) licensed and fully mobile unlicensed users. Examples of real networks that can be modeled like this include Wi-Fi, cellular and mesh networks with both licensed and unlicensed users. In such networks, the licensed users may not schedule their transmissions and thus send at any time they desire. The unlicensed users must make an effort to opportunistically use the spectrum holes without interfering too much with the licensed users, and hence need sophisticated scheduling mechanisms.

A taxonomy of different approaches to spectrum sharing in cognitive networks is provided in [GJMS09, Bud07, ZS07]. The network model used in this chapter falls into the "interweave" approach to spectrum sharing.

2.3 Maximum Throughput Objective

Let r_n denote the time average rate of admitted data for secondary user n, i.e.,

$$r_n = \lim_{t\to\infty} \frac{1}{t} \sum_{\tau=0}^{t-1} R_n(\tau)$$

Let $\boldsymbol{r} = (r_1, \ldots, r_N)$ denote the vector of these time average rates.

We define the following "collision" variables for each primary user $m \in \{1, \ldots, M\}$:

$$C_m(t) = \begin{cases} 1 & \text{if there was a collision with primary user in channel } m \text{ in slot } t \\ 0 & \text{else} \end{cases}$$

Let c_m denote the time average rate of collision for primary user m, i.e.,

$$c_m = \lim_{t\to\infty} \frac{1}{t} \sum_{\tau=0}^{t-1} C_m(\tau)$$

Let $\{\theta_1, \ldots, \theta_N\}$ be a collection of positive weights. Then the control objective is to design an admission control and scheduling policy that yields time average rate vector $\boldsymbol{r}$ that solves the following optimization problem:

$$\begin{aligned} \text{Maximize:} \quad & \sum_{n=1}^{N} \theta_n r_n \\ \text{Subject to:} \quad & 0 \le r_n \le \lambda_n \; \forall \, n \in \{1, \ldots, N\} \\ & c_m \le \rho_m \nu_m \; \forall \, m \in \{1, \ldots, M\} \\ & \boldsymbol{r} \in \Lambda \end{aligned}$$

Here, Λ represents the *network capacity region* for the network model as described above. It is defined as the set of all input rate vectors $\vec{\lambda} = (\lambda_1, \ldots, \lambda_N)$ of the secondary users for which a scheduling strategy exists that can support $\vec{\lambda}$ (without admission control) subject to the constraints imposed by the network. The notion of network capacity for general networks with time varying channels and energy constraints is formalized in [NMR05, Nee06, GNT06] where it is shown to be a function of the steady state network topology distribution, channel probabilities, and time average transmission rates.

Let $\boldsymbol{r}^* = (r_1^*, \ldots, r_N^*)$ denote the optimal solution to the optimization problem defined above. In principle, it can be solved if all system parameters are known in advance including Λ. However, in practice, this region may not be known to the network controller (e.g., because the mobility patterns of the secondary users are unknown) and the above maximization problem must be done for input rates either inside or outside of the capacity region. Even if all system parameters are known, the optimal solution may be difficult to implement as it may require centralized coordination among all users.

We next present an online control algorithm that overcomes all of these challenges.

2.4 Optimal Control Algorithm

We now present the *Cognitive Network Control Algorithm* (CNC), a cross-layer control strategy that can be shown to achieve the optimal solution $\boldsymbol{r}^*$ to the network optimization problem presented earlier. It operates without knowledge of whether the input rate is within or outside of the capacity region Λ. Further, it provides *deterministic worst case* bounds on the maximum secondary user queue backlog at all times and the maximum number of collisions with a primary user in a given time interval. These are much stronger than probabilistic performance guarantees. Finally, it offers a control parameter V that enables an explicit trade-off between the average throughput utility and delay. This algorithm is similar in spirit to the "backpressure" algorithms proposed in [Nee06, NU07] for problems of energy optimal networking in wireless ad-hoc and mesh networks.

The algorithm is decoupled into two separate components. The first component performs optimal admission control at the transport layers and is implemented independently at each secondary user. The second component determines a network wide resource allocation every slot and needs to be solved collectively by the secondary users.

In addition to the actual queue backlog $Q_n(t)$, this algorithm uses a set of collision queues $X_m(t)$ for each channel m. These queues are "virtual" in that they are maintained purely in software. These are used to track the amount by which the number of collisions suffered by a primary user m exceeds its time average collision fraction ρ_m. These could be maintained at the primary user base station for each channel. We assume that the secondary users are aware of the $X_m(t)$ value for each channel m that they can access at time t. We define the collision queue $X_m(t)$ for channel m as follows:

$$X_m(t+1) = \max[X_m(t) - \rho_m 1_m(t), 0] + C_m(t) \tag{2.8}$$

where $C_m(t)$ is the collision variable for channel m as defined in the previous section and $1_m(t)$ is an indicator variable, taking value 1 if primary user m transmits in slot t and 0 else (so that $1_m(t) = 1 - S_m(t)$). The above equation represents the queueing dynamics of a single server system with input process $C_m(t)$ and service process $\rho_m 1_m(t)$. This system is stable only when the service rate is greater than or equal to the input rate, i.e.,

$$c_m = \lim_{t\to\infty} \frac{1}{t} \sum_{\tau=0}^{t-1} C_m(\tau) \leq \lim_{t\to\infty} \rho_m \frac{1}{t} \sum_{\tau=0}^{t-1} 1_m(\tau) = \rho_m \nu_m$$

This is precisely the collision constraint in the utility optimization problem stated earlier. *Thus, if our policy stabilizes all collision queues as defined above, the maximum average rate of collisions will meet the required constraint.* This technique of turning time average constraints into queueing stability problems was introduced in [Nee06] where it was used for satisfying average power constraints.

2.4.1 Cognitive Network Control Algorithm (CNC)

Let $V \geq 0$ be a fixed control parameter. Let the admission control and resource allocation decision under the CNC algorithm be $R_n^{CNC}(t)$ and $\mu_{nm}^{CNC}(t)$ respectively. These are determined as follows:

1. *Admission Control:* At each secondary user n, choose the number of packets to admit $R_n^{CNC}(t)$ as the solution to the following problem:

$$\begin{aligned} &\text{Minimize:} \quad && R_n(t)[Q_n(t) - V\theta_n] \\ &\text{Subject to:} \quad && 0 \leq R_n(t) \leq A_n(t) \end{aligned} \tag{2.9}$$

This problem has a simple threshold-based solution. In particular, if the current queue backlog $Q_n(t) > V\theta_n$, then $R_n^{CNC}(t) = 0$ and no new packets are admitted. Else, if $Q_n(t) \leq V\theta_n$, then $R_n^{CNC}(t) = A_n(t)$ and all new packets are admitted. Note that this can be solved separately at each user and does not require knowledge of θ_n weights of other users.

2. *Resource Allocation:* Choose a resource allocation $\mu_{nm}^{CNC}(t)$ that solves the following problem:

$$\begin{aligned} &\text{Maximize:} \quad && \sum_{n,m} \mu_{nm}(t)\Big[Q_n(t)P_m(t) - \sum_{k=1}^{M} X_k(t)(1 - P_k(t))I_{nm}^k(t)\Big] \\ &\text{Subject to:} \quad && \text{constraints (2.2), (2.3), (2.4), (2.5)} \end{aligned} \tag{2.10}$$

After observing the outcome of this allocation at the end of the slot, the virtual queues are updated as in (2.8) based on the feedback received about a collision with a primary user or a successful transmission. Note that only collisions with a primary user affect (2.8), collisions between secondary users do not affect the virtual collision queues.

The above problem is a generalized Maximum Weight Match problem where the weight for a pair (n, m) is given by $\left(Q_n(t)P_m(t) - \sum_{k=1}^{M} X_k(t)(1 - P_k(t))I_{nm}^k(t)\right)$. This is the difference between the current queue backlog $U_n(t)$ weighted by the probability that primary user m is idle and the weighted sum of all collision queue backlogs $X_k(t)$ for the channels that user n interferes with if it uses channel m. The weight for a collision queue is the probability that the corresponding primary user will transmit. Note that if this difference is non-positive, then the link (n, m) can be removed from the decision options, simplifying scheduling. This problem is hard to solve in general, though constant factor approximations exist that are easier to implement. We discuss these in Sec. 2.6.

For the case when all channels are orthogonal from the point of view of secondary users (which means a secondary user transmission on a channel does not cause interference to other channels), $\mathcal{I}_{nm}(t) = \{m\}$ so that $I_{nm}^m(t) = 1, I_{nm}^k(t) = 0 \;\forall\; k \neq m$. Then the above maximization simplifies to the following problem:

$$
\begin{aligned}
\text{Maximize:} \quad & \sum_{n,m} \mu_{nm}(t)\Big[Q_n(t)P_m(t) - X_m(t)(1 - P_m(t))\Big] \\
\text{Subject to:} \quad & \text{constraints } (2.2), (2.3), (2.4), (2.7)
\end{aligned} \tag{2.11}
$$

The above maximization requires solving the Maximum Weight Match (MWM) problem on an $N \times M$ bipartite graph of N secondary users and M channels. This problem can be solved in polynomial time, though this may require centralized control. We discuss simpler constant factor approximations in Sec. 2.6. Also, we consider a cell partitioned network in the simulations of Sec. 2.7 for which a full maximum weight match can be implemented in a distributed manner.

To get an intuition behind the algorithm, consider the maximization in (2.11) for the orthogonal channel case. A secondary user n would attempt transmission over channel m only if $Q_n(t)P_m(t) > X_m(t)(1 - P_m(t))$. Intuitively, this algorithm tries to schedule secondary users with larger queue backlogs over those channels that are more likely to be idle and that have smaller "effective" collision queue values. Here, the effective collision queue value is its actual value weighted by the probability of that channel being busy with its primary user. Intuitively, these collision queues enable stochastic optimization by acting as dynamic Lagrange multipliers [GNT06]. Using (2.11), the dynamic weights of $X_m(t)$ help determine the best channel for attempting transmission.

2.4.2 Comparison with a Counter Based Algorithm

The virtual collision queues $X_m(t)$ play a crucial role in making optimal control decisions. To illustrate this, we compare the performance of CNC with a Counter Based Algorithm on a simple example network with one static secondary user and two primary channels. In this algorithm, a count of the number of collisions suffered so far is maintained for each primary channel. In each slot, a channel m is considered eligible for access only if the average rate of collisions so far does not violate the constraint $\rho_m \nu_m$. Further, if both the channels are eligible, then the algorithm selects the one that is more likely to be idle. Note that unlike CNC, this algorithm does not make use of the queue values (real or virtual) in making control decisions.

In the example we consider, we assume that both primary channels evolve independently according to the 2 state Markov Chain of Fig. 2.2 with $P_{10} = \epsilon = 1/3$ and

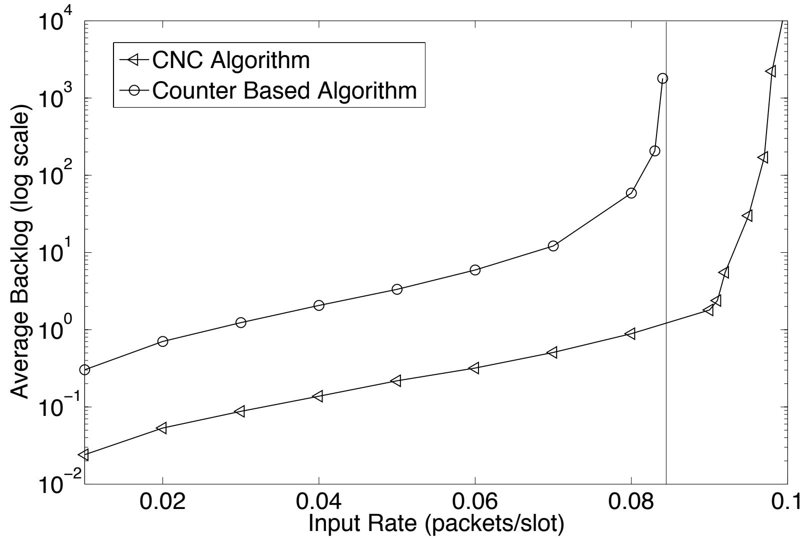

Figure 2.3: Total average congestion vs. input rate under the Counter Based Algorithm and CNC.

$P_{01} = \delta = 1/3$. This means that $\nu_1 = \nu_2 = 0.5$ packets/slot. We assume that the maximum collision fraction $\rho_m = 0.05$ for both channels, so that for each primary user, at most 5% of its packets can have collisions.

New packets arrive at the secondary user according to an i.i.d. Bernoulli process of rate λ. For simplicity, we assume no admission control so that all arrivals are accepted into the network queue. In Fig. 2.3, we plot the average congestion at the secondary user under the Counter Based Algorithm and CNC for different values of the input rate λ. The vertical lines in Fig. 2.3, which appear at $\lambda = 0.085$ packets/slot and $\lambda = 0.1$ packets/slot, represent the maximum secondary throughput achieved under these algorithms. From this, it can be seen that CNC significantly outperforms the Counter Based Algorithm. Intuitively, this is because the Counter Based Algorithm is more conservative than CNC. Unlike the Counter Based Algorithm, under CNC, a channel m may be accessed even if the average rate of collisions seen by it so far temporarily violates the constraint $\rho_m \nu_m$.

For this simple example, we can also compute the optimal solution exactly using linear programming. There are 4 possible values of the cumulative channel state in the last slot $(S_1(t-1), S_2(t-1))$ given by $(0,0), (0,1), (1,0)$, and $(1,1)$. Let this set be denoted by $\mathcal{S}$. For each $i \in \mathcal{S}$, let $x_{i,m}$ be the probability that the secondary user transmits on channel m in slot t (where $m \in \{1,2\}$) *given that* the cumulative channel state in the last slot was i. For example, $x_{(0,0),1}$ is the probability that the secondary user transmits on channel 1 in slot t given that $(S_1(t-1), S_2(t-1)) = (0,0)$. Using this, the problem of maximizing the secondary user throughput subject to the time average collision constraints can be written as the following linear program:

$$\begin{aligned}
\text{Maximize:} \quad & \pi_{(0,0)}[x_{(0,0),1}P_{01} + x_{(0,0),2}P_{01}] + \pi_{(0,1)}[x_{(0,1),1}P_{01} + x_{(0,1),2}P_{11}] \\
& + \pi_{(1,0)}[x_{(1,0),1}P_{11} + x_{(1,0),2}P_{01}] + \pi_{(1,1)}[x_{(1,1),1}P_{11} + x_{(1,1),2}P_{11}] \qquad (2.12) \\
\text{Subject to:} \quad & \pi_{(0,0)}[x_{(0,0),1}P_{00}] + \pi_{(0,1)}[x_{(0,1),1}P_{00}] \\
& + \pi_{(1,0)}[x_{(1,0),1}P_{10}] + \pi_{(1,1)}x_{(1,1),1}P_{10} \leq \rho_1(\pi_{(0,0)} + \pi_{(0,1)}) \qquad (2.13) \\
& \pi_{(0,0)}[x_{(0,0),2}P_{00}] + \pi_{(0,1)}[x_{(0,1),2}P_{10}] \\
& + \pi_{(1,0)}[x_{(1,0),2}P_{00}] + \pi_{(1,1)}x_{(1,1),2}P_{10} \leq \rho_2(\pi_{(0,0)} + \pi_{(1,0)}) \qquad (2.14) \\
& 0 \leq x_{i,m} \leq 1 \ \forall i \in \mathcal{S}, m \in \{1,2\}
\end{aligned}$$

where π_i denotes the steady-state probability of being in state $i \in \mathcal{S}$ and $P_{00} = P_{11} = \frac{2}{3}, P_{10} = P_{01} = \frac{1}{3}$ denote the transition probabilities of the 2 state Markov Chain of Fig. 2.2. The objective in (2.12) represents the expected secondary user throughput under this randomized policy. To see this, consider the first term $x_{(0,0),1}P_{01}$ in (2.12). This is the probability that the secondary user transmits on channel 1 and channel 1 transitions to state 1 (idle) in the current slot *given that* both channels were in state 0 (busy) in the last slot. The other terms can be obtained similarly. (2.13) and (2.14) represent the time-average rate of collisions seen by the primary channels 1 and 2. For example, the first term $x_{(0,0),1}P_{00}$ in (2.13) is the probability that the secondary user transmits on channel 1 and channel 1 transitions to state 0 (busy) in the current slot *given that* both channels were in state 0 (busy) in the last slot. The other terms can be obtained similarly.

By solving this linear program, we obtain the maximum throughput as 0.1 packets/slot. Thus, the CNC algorithm is able to achieve the maximum throughput as V is increased.

2.4.3 Performance Analysis

We now characterize the performance of the CNC algorithm. This holds for general secondary user mobility processes that are described by finite state ergodic Markov Chains.

Theorem 1 *(CNC Algorithm Performance) Assume that all queues are initialized to* 0. *Suppose all arrivals* $A_n(t)$ *are upper bounded so that* $A_n(t) \leq A_{max}$ *for all* n, t. *Also suppose the* $\boldsymbol{H}(t)$ *and* $\boldsymbol{P}(t)$ *processes are Markovian and have a well defined steady state distribution. Then, implementing the* CNC *algorithm every slot for any fixed control parameter* $V \geq 0$ *stabilizes all real and virtual queues (thereby satisfying the maximum time average collision constraints) and yields the following performance bounds:*

1. *The worst case queue backlog for each secondary user n is upper bounded by a finite constant $Q_{n,max}$ for all t:*

$$Q_n(t) \leq Q_{n,max} \triangleq V\theta_n + A_{max} \tag{2.15}$$

Let $\theta_{max} = \max_{n \in \{1,\ldots,N\}}\{\theta_n\}$. Then, from (2.15) we have for any n

$$Q_n(t) \leq Q_{max} \triangleq V\theta_{max} + A_{max} \tag{2.16}$$

2. *For all m, t such that $P_m(t) \neq 1$, let $\epsilon > 0$ be such that $P_m(t) \leq 1 - \epsilon$.*[1] *Then, the worst case collision queue backlog for all channels m is upper bounded by a finite constant X_{max}:*

$$X_m(t) \leq X_{max} \triangleq Q_{max}\frac{(1-\epsilon)}{\epsilon} + 1 \tag{2.17}$$

Further, the worst case number of collisions suffered by any primary user m is no more than $\rho_m T + X_{max}$ over any interval (of size greater than or equal to T slots) over which the primary user transmits T times, for any positive integer T.

3. *The time average throughput utility achieved by the CNC algorithm is within $\tilde{B}/V$ of the optimal value:*

$$\liminf_{t\to\infty} \frac{1}{t}\sum_{\tau=0}^{t-1}\sum_{n=1}^{N}\theta_n \mathbb{E}\left\{R_n(\tau)\right\} \geq \sum_{n=1}^{N}\theta_n r_n^* - \frac{\tilde{B}}{V} \tag{2.18}$$

[1] Such an ϵ exists for any finite state ergodic Markov Chain.

where $\tilde{B} = B + C_U + C_X + N + M$ and where B, C_U, C_X are constants (defined precisely in (2.21), (A.2), (A.3)).

The constants C_U and C_X are determined by the stochastics of the mobility and channel state probability processes and it is shown in Appendix A.1 that these are $O(\log V)$ when these processes evolve according to any finite state ergodic Markov model. Therefore, by part (3) of the theorem, the achieved average throughput utility is within $O(\log V/V)$ of the optimal value. This can be pushed arbitrarily close to the optimal value by increasing the control parameter V. However, this increases the maximum queue backlog bound Q_{max} linearly in V, leading to a utility-delay trade-off.

The above bounds are quite strong. In particular, the maximum collisions bound in part (2) gives deterministic performance guarantees that hold for *any* interval size. This is quite useful in the context of cognitive networks since it implies that the licensed users are guaranteed to suffer at most these many collisions. Probabilistic guarantees (e.g., [CZS08], [HLD08]) do not provide such bounds.

We next prove the first two parts of Theorem 1. Proof of part (3) uses the technique of Stochastic Lyapunov Optimization and is provided in the next section.

Proof 1 *(Proof of part (1)): Suppose that $Q_n(t) \leq Q_{n,max}$ for all $n \in \{1, \ldots, N\}$ for some time t. This is true for $t = 0$ as all queues are initialized to* 0*. We show that the same holds for time $t+1$. We have 2 cases. If $Q_n(t) \leq Q_{n,max} - A_{max}$, then from (2.1), we have $Q_n(t+1) \leq Q_{n,max}$ (because $R_n(t) \leq A_{max}$ for all t). Else, if*

$Q_n(t) > Q_{n,max} - A_{max}$, then $Q_n(t) > V\theta_n + A_{max} - A_{max} = V\theta_n$. Then, the admission control part of the algorithm chooses $R_n(t) = 0$, so that by (2.1):

$$Q_n(t+1) \leq Q_n(t) \leq Q_{n,max}$$

This proves (2.15).

(Proof of part (2)): Suppose that $X_m(t) \leq X_{max}$ for all $m \in \{1, \ldots, M\}$ for some time t. This is true for $t = 0$ as all queues are initialized to 0. We show that the same holds for time $t + 1$. First suppose $P_m(t) = 1$. Then, by definition, there is no collision with the primary user in channel m in slot t so that $C_m(t) = 0$. Then, from (2.8), we have $X_m(t+1) \leq X_{max}$. Next, suppose $P_m(t) < 1$. We again have 2 cases. If $X_m(t) \leq X_{max} - 1$, then from (2.8), we have $X_m(t+1) \leq X_{max}$ (because $C_m(t) \leq 1$ for all t). Else, if $X_m(t) > X_{max} - 1 = Q_{max}\frac{(1-\epsilon)}{\epsilon}$, then $X_m(t)\epsilon > Q_{max}(1-\epsilon)$. This implies $X_m(t)(1 - P_m(t)) \geq X_m(t)\epsilon > Q_{max}(1-\epsilon) \geq Q_{max}P_m(t) \geq Q_n(t)P_m(t)$ for all $n \in \{1, \ldots, N\}$. Thus, the resource allocation part of the algorithm chooses $\mu_{nm}(t) = 0$ for all n. This would yield $C_m(t) = 0$ (since no collision takes place with primary user m), so that by (2.8):

$$X_m(t+1) \leq X_m(t) \leq X_{max}$$

This proves (2.17).

Now consider any interval (t_1, t_2) *in which primary user* m *transmits* T *times. Then, from the queueing equation (2.8) we have that:*

$$X_m(t_2 + 1) \geq X_m(t_1) + \sum_{\tau=t_1}^{t_2} C_m(\tau) - \rho_m T$$

This follows by noting that $\rho_m T$ *is the maximum number of "departures" that can take place in the queueing dynamics (2.8) during the interval* (t_1, t_2)*. From this, we can bound the worst case number of collisions suffered by primary user* m *over any interval in which it transmits* T *times as:*

$$\sum_{\tau=t_1}^{t_2} C_m(\tau) \leq \rho_m T + X_{max}$$

2.5 Stochastic Lyapunov Optimization

Let $\boldsymbol{Q}(t) = (Q_1(t), \ldots, Q_K(t))$ be a vector process of queue lengths for a discrete time stochastic queueing network with K queues (possibly including some virtual queues like the collision queues defined in the previous subsection). Let $L(\boldsymbol{Q})$ be any non-negative scalar valued function of the queue lengths, called a Lyapunov function. Define the Lyapunov drift $\Delta(t)$ as follows:

$$\Delta(t) \triangleq \mathbb{E}\left\{L(\boldsymbol{Q}(t+1)) - L(\boldsymbol{Q}(t))\right\}$$

Suppose the network accumulates "rewards" every timeslot (where rewards might correspond to utility measures of control actions). Assume rewards are real valued and bounded, and let the stochastic process $f(t)$ represent the reward earned during slot t. Let f^* represent the target reward. The following result (a variant of related results from [Nee06, GNT06]) specifies a drift condition which ensures that the time average of the reward process $f(t)$ is close to meeting or exceeding f^*.

Theorem 2 *(Delayed Lyapunov optimization with Rewards) Suppose there exist finite constants $V > 0, B > 0, d > 0$, and a non-negative function $L(\boldsymbol{Q})$ such that $\mathbb{E}\{L(\boldsymbol{Q}(d))\} < \infty$ and for every timeslot $t > d$, the Lyapunov drift satisfies:*

$$\Delta(t) - V\mathbb{E}\{f(t)\} \leq B - Vf^* \tag{2.19}$$

then we have:

$$\liminf_{t\to\infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}\{f(\tau)\} \geq f^* - \frac{B}{V}$$

Proof 2 *Inequality (2.19) holds for all $t > d$. Summing both sides over $\tau \in \{d, \dots, t-1\}$ yields:*

$$\mathbb{E}\{L(\boldsymbol{Q}(t))\} - \mathbb{E}\{L(\boldsymbol{Q}(d))\} \leq B(t-d) - V(t-d)f^* + V\sum_{\tau=d}^{t-1} \mathbb{E}\{f(\tau)\}$$

Rearranging terms, dividing by t, and using non-negativity of $L(\boldsymbol{Q})$ yields:

$$\frac{(t-d)f^*}{t} - \frac{(t-d)B}{tV} - \frac{\mathbb{E}\{L(\boldsymbol{Q}(d)\}}{tV} \leq \frac{1}{t}\sum_{\tau=0}^{t-1} \mathbb{E}\{f(\tau)\}$$

The result follows by taking limit as $t \to \infty$.

We now use Theorem 2 to prove part (3) of Theorem 1. This is done by comparing the Lyapunov drift of the CNC algorithm with that of a stationary randomized algorithm $STAT$ that makes control decisions every slot purely as a function of the current channel state information $\boldsymbol{P}(t)$ and $\boldsymbol{H}(t)$.

We first obtain an expression for the Lyapunov drift under any control policy for our cognitive network model.

2.5.1 Lyapunov Drift

Let $\boldsymbol{Q}(t) = (Q_1(t), \ldots, Q_N(t), X_1(t), \ldots, X_M(t))$ represent the collection of all real and virtual queue backlogs in the cognitive network. We define the following Lyapunov function:

$$L(\boldsymbol{Q}(t)) \triangleq \frac{1}{2}\Big[\sum_{n=1}^{N} Q_n^2(t) + \sum_{m=1}^{M} X_m^2(t)\Big]$$

Using queueing dynamics (2.1) and (2.8), the Lyapunov drift $\Delta(t)$ under any control policy (including CNC) can be computed as follows:

$$\begin{aligned}\Delta(t) \leq & B - \mathbb{E}\left\{\sum_{n=1}^{N} Q_n(t)\Big(\sum_{m=1}^{M} \mu_{nm}(t) S_m(t) - R_n(t)\Big)\right\} \\ & - \mathbb{E}\left\{\sum_{m=1}^{M} X_m(t)(\rho_m 1_m(t) - C_m(t))\right\} \end{aligned} \tag{2.20}$$

where

$$B \triangleq \frac{N(A_{max}^2 + 1) + \sum_{m=1}^{M} \rho_m^2 + M}{2} \tag{2.21}$$

The collision variable $C_m(t)$ can be expressed in terms of the control decisions $\mu_{ij}(t)$ and channel state $\boldsymbol{S}(t)$ as follows:

$$C_m(t) = \sum_{i=1}^{N} \sum_{j=1}^{M} \mu_{ij}(t) I_{ij}^m(t) 1_{[U_i(t)>0]}(1 - S_m(t)) \tag{2.22}$$

where $1_{[U_i(t)>0]}$ is an indicator variable of non-zero queue backlog in secondary user i. This follows by observing that a collision with the primary user occurs in channel m if the primary user is busy (i.e. $S_m(t) = 0$) and if $\mu_{ij}(t) = 1$ for some secondary user i with non-zero backlog using channel j that interferes with channel m. We will find it useful to define the following related variable:

$$\hat{C}_m(t) = \sum_{i=1}^{N} \sum_{j=1}^{M} \mu_{ij}(t) I_{ij}^m(t)(1 - S_m(t)) \tag{2.23}$$

For a given control parameter $V \geq 0$, we subtract the reward metric $V\mathbb{E}\left\{\sum_{n=1}^{N} \theta_n R_n(t)\right\}$ from both sides of the drift inequality (2.20) and use the fact that $\hat{C}_m(t) \geq C_m(t)\ \forall t$ to get the following:

$$\begin{aligned} \Delta(t) - V\mathbb{E}\left\{\sum_{n=1}^{N} \theta_n R_n(t)\right\} &\leq B - \mathbb{E}\left\{\sum_{n=1}^{N} Q_n(t)\Big(\sum_{m=1}^{M} \mu_{nm}(t) S_m(t) - R_n(t)\Big)\right\} \\ - \mathbb{E}\left\{\sum_{m=1}^{M} X_m(t)(\rho_m 1_m(t) - \hat{C}_m(t))\right\} &- V\mathbb{E}\left\{\sum_{n=1}^{N} \theta_n R_n(t)\right\} \end{aligned} \tag{2.24}$$

2.5.2 Optimal Stationary, Randomized Policy

We now describe the stationary, randomized policy $STAT$ that chooses control actions only as a function of $\boldsymbol{P}(t)$ and $\boldsymbol{H}(t)$ every slot. We have the following lemma:

Lemma 1 *(Optimal Stationary, Randomized Policy): For any rate vector* $(\lambda_1, \ldots, \lambda_N)$ *(inside or outside of the network capacity region* Λ*), there exists a stationary randomized scheduling policy* $STAT$ *that chooses feasible allocations* $R_n^{STAT}(t), \mu_{nm}^{STAT}(t)$ *for all* $n \in \{1, \ldots, N\}, m \in \{1, \ldots, M\}$ *every slot as a function of the channel state information* $\boldsymbol{P}(t)$ *and* $\boldsymbol{H}(t)$ *and yields the following steady state values:*

$$\mathbb{E}\left\{R_n^{STAT}(t)\right\} = r_n^* \;\forall\; t \tag{2.25}$$

$$\mu_n^{STAT} \triangleq \lim_{t\to\infty} \frac{1}{t}\sum_{\tau=0}^{t-1} \mathbb{E}\left\{\sum_{m=1}^{M} \mu_{nm}^{STAT}(\tau) S_m(\tau)\right\} \geq r_n^* \tag{2.26}$$

$$\hat{c}_m^{STAT} \triangleq \lim_{t\to\infty} \frac{1}{t}\sum_{\tau=0}^{t-1} \mathbb{E}\left\{\hat{C}_m^{STAT}(t)\right\} \leq \rho_m \nu_m \tag{2.27}$$

Specifically, the admission control decision $R_n^{STAT}(t)$ under this policy is determined as follows. At each secondary user n, observe $A_n(t)$ and choose $R_n(t)^{STAT}$ as follows:

$$R_n^{STAT}(t) = \begin{cases} A_n(t) & \text{with probability } r_n^*/\lambda_n \\ 0 & \text{else} \end{cases}$$

These probabilistic decisions are made every slot independent of the current queue backlogs and are i.i.d with probability $r_n^*/\lambda_n \leq 1$. Thus, we have

$$\mathbb{E}\left\{R_n(t)^{STAT}\right\} = \mathbb{E}\left\{A_n(t)\right\}\frac{r_n^*}{\lambda_n} = r_n^*$$

The above facts can be proven using techniques similar to the ones used in [NMR05, NML08, Nee06] for showing the existence of capacity achieving stationary, randomized policies that make control decisions independent of queue backlog. We now prove an important property of the CNC algorithm.

Claim: Suppose the CNC algorithm is implemented on all slots up to time t. Thus, the queue backlogs $U_n(t)$ and $X_m(t)$ are determined by the history before time t and are not affected by the control decisions made on slot t. Then, given the current queue backlogs, the CNC control decisions for slot t minimize the right hand side of inequality (2.24) over all alternative feasible policies that could be implemented on slot t, including the stationary, randomized policy $STAT$.

Note that we are not claiming that the CNC policy, implemented over time, minimizes the right hand side expectation of (2.24) at time t. Indeed, another policy may result in a smaller expected queue size outcome at time t. Rather, we are claiming that, given CNC is used up to (but not including) time t (so that queue sizes at time t are already determined by the sample path outcome of CNC up to this time), the CNC control decisions made *at time* t act to greedily minimize the right hand side over any other decisions that can be made at time t.

Proof: By changing the order of summations and using (2.23), the right side of (2.24) can be expressed in a more convenient form:

$$\begin{aligned} & B - \sum_{m=1}^{M} \rho_m \mathbb{E}\left\{X_m(t)1_m(t)\right\} + \mathbb{E}\left\{\sum_{n=1}^{N} R_n(t)(Q_n(t) - V\theta_n)\right\} \\ & - \mathbb{E}\left\{\sum_{n,m} \mu_{nm}(t)\Big[Q_n(t)S_m(t) - \sum_{k=1}^{M} X_k(t)(1 - S_k(t))I_{nm}^k\Big]\right\} \end{aligned} \quad (2.28)$$

where we have omitted the t subscript in $I_{nm}^k(t)$. Note that $\mathbb{E}\left\{S_m(t)|\chi(t)\right\} = Pr[S_m(t) = 1|\chi(t)] = P_m(t)\ \forall m$. By writing the last two terms on the right hand side as an iterated expectation by conditioning on the queue backlog and $\chi(t)$, it can be seen that CNC chooses control decisions (2.9) and (2.10) that minimize these terms for every possible value of the backlog and $\chi(t)$, so that the actual expectation is also minimized. We note that the unconditioning is done with respect to the queue backlog distribution that arises as a result of implementing the CNC algorithm for all slots up to time t. Using this fact, we have:

$$\begin{aligned} & \Delta^{CNC}(t) - V\mathbb{E}\left\{\sum_{n=1}^{N} \theta_n R_n^{CNC}(t)\right\} \leq B - \mathbb{E}\left\{\sum_{n=1}^{N} Q_n(t)\Big(\sum_{m=1}^{M} \mu_{nm}^{STAT}(t)S_m(t) - R_n^{STAT}(t)\Big)\right\} \\ & - \mathbb{E}\left\{\sum_{m=1}^{M} X_m(t)(\rho_m 1_m(t) - \hat{C}_m^{STAT}(t))\right\} - V\mathbb{E}\left\{\sum_{n=1}^{N} \theta_n R_n^{STAT}(t)\right\} \end{aligned} \quad (2.29)$$

In Appendix A.1, we show that for all $t > d$ (where d is a finite positive integer and is computed in Appendix A.1), this can be expressed as:

$$\Delta^{CNC}(t) - V\mathbb{E}\left\{\sum_{n=1}^{N} \theta_n R_n^{CNC}(t)\right\} \leq \tilde{B} - V\sum_{n=1}^{N} \theta_n r_n^* \quad (2.30)$$

This is in a form that fits (2.19). Thus, applying Theorem 2 proves (2.18).

2.6 Distributed Implementation

Here we discuss constant factor approximations to the resource allocation problem (2.10) that are easier to implement in a distributed network. We focus on the orthogonal channel case in which a secondary user transmission on a channel does not cause interference to other channels. As noted earlier, in this case, the resource allocation problem (2.11) reduces to a Maximum Weight Match (MWM) problem on an $N \times M$ bipartite graph between N secondary users and M channels. An edge exists between nodes n and m of this graph if $h_{nm}(t) = 1$, i.e., if secondary user n can access channel m in slot t. The weight of this edge is given by $(Q_n(t)P_m(t) - X_m(t)(1 - P_m(t)))$. While the MWM problem can be solved in polynomial time in a centralized way, here we are interested in simpler implementations. In particular, we use the idea of *Greedy Maximal Weight Match Scheduling* that has been investigated in several recent works including [LS06, CKLS08, WSP07].

A maximal match is defined as any set of edges (m, n) that do not interfere with each other such that adding any new edge to this set necessarily violates a matching constraint. A Greedy Maximal Weight Match can be achieved as follows: First select the edge (m, n) with the largest positive weight and label it "active". Then select the edge with the second largest positive weight (breaking ties arbitrarily) that does not conflict with an active edge and label it active. Continue in the same way, until no more edges can be added. It is not difficult to see that this final set of edges labeled "active" has the desired maximal property. A Greedy Maximal Weight Match can be computed with much less overhead as compared to the Maximum Weight Match.

It can be shown that using such greedy maximal weight matches instead of the maximum weight match every slot can still support any rate within $\frac{1}{2}\Lambda$. In particular, in Appendix A.3, we show that resource allocation $\mu_{nm}^{GMM}(t)$ chosen according to a Greedy Maximal Weight Match has the following property:

$$\sum_{n,m} \mu_{nm}^{GMM}(t)\Big[Q_n(t)P_m(t) - X_m(t)(1 - P_m(t))\Big] \geq$$
$$\frac{1}{2}\sum_{n,m} \mu_{nm}^{CNC}(t)\Big[Q_n(t)P_m(t) - X_m(t)(1 - P_m(t))\Big] \qquad (2.31)$$

where $\mu_{nm}^{CNC}(t)$ is the optimal solution to (2.11). Using this, we get the following result:

Theorem 3 *(Performance Bound for Orthogonal Channels with Greedy Maximal Weight Match Scheduling) The time average throughput utility achieved by the CNC algorithm with Greedy Maximal Weight Match Scheduling is within* $\frac{B^{GMM}}{V}$ *of* $\frac{1}{2}\sum_{n=1}^{N}\theta_n r_n^*$*:*

$$\liminf_{t\to\infty} \frac{1}{t}\sum_{\tau=0}^{t-1}\sum_{n=1}^{N} \theta_n \mathbb{E}\left\{R_n(\tau)\right\} \geq \frac{1}{2}\sum_{n=1}^{N}\theta_n r_n^* - \frac{B^{GMM}}{V} \qquad (2.32)$$

where $B^{GMM} = (\tilde{B} + B)/2$.

We note that while using Greedy Maximal Weight Match Scheduling provides a factor of 2 approximation in terms of the time average throughput utility, the deterministic bounds on maximum queue backlog and worst case number of collisions remain the same as in parts (1) and (2) of Theorem 1. This is because the arguments there were based only on the fact that only positive weight transmissions are scheduled, which also holds for GMM.

Proof 3 *Let $R_n^{GMM}(t)$ and $\mu_{nm}^{GMM}(t)$ denote the admission control and resource allocation decisions under Greedy Maximal Match Scheduling. Let $\Delta^{GMM}(t)$ be the corresponding Lyapunov drift. Note that for any given queue backlog $\boldsymbol{Q}(t)$, $R_n^{GMM}(t) = R_n^{CNC}(t)$. Then, using (2.28), we have:*

$$\Delta^{GMM}(t) - V\mathbb{E}\left\{\sum_{n=1}^{N} \theta_n R_n^{GMM}(t)\right\} \leq B - \sum_{m=1}^{M} \rho_m \mathbb{E}\left\{X_m(t) 1_m(t)\right\}$$
$$+ \mathbb{E}\left\{\sum_{n=1}^{N} R_n^{GMM}(t)(Q_n(t) - V\theta_n)\right\} - \mathbb{E}\left\{\sum_{n,m} \mu_{nm}^{GMM}(t)\Big[Q_n(t)S_m(t) - X_m(t)(1 - S_m(t))\Big]\right\}$$

Using property (2.31) and the fact that $R_n^{GMM}(t) = R_n^{CNC}(t)$, the above can be written as:

$$\Delta^{GMM}(t) - V\mathbb{E}\left\{\sum_{n=1}^{N} \theta_n R_n^{GMM}(t)\right\} \leq B - \sum_{m=1}^{M} \rho_m \mathbb{E}\left\{X_m(t) 1_m(t)\right\}$$
$$+ \mathbb{E}\left\{\sum_{n=1}^{N} R_n^{CNC}(t)(Q_n(t) - V\theta_n)\right\} - \frac{1}{2}\mathbb{E}\left\{\sum_{n,m} \mu_{nm}^{CNC}(t)\Big[Q_n(t)S_m(t) - X_m(t)(1 - S_m(t))\Big]\right\}$$

From (2.9), note that $R_n^{CNC}(t) \geq 0$ if $Q_n(t) \leq V\theta_n$, else $R_n^{CNC}(t) = 0$. Therefore the second to last term under the admission control of CNC is non-positive. Thus, the above can be rewritten as:

$$\Delta^{GMM}(t) - V\mathbb{E}\left\{\sum_{n=1}^{N} \theta_n R_n^{GMM}(t)\right\} \leq B - \frac{1}{2}\sum_{m=1}^{M} \rho_m \mathbb{E}\left\{X_m(t) 1_m(t)\right\}$$
$$+ \frac{1}{2}\mathbb{E}\left\{\sum_{n=1}^{N} R_n^{CNC}(t)(Q_n(t) - V\theta_n)\right\} - \frac{1}{2}\mathbb{E}\left\{\sum_{n,m} \mu_{nm}^{CNC}(t)\Big[Q_n(t)S_m(t) - X_m(t)(1 - S_m(t))\Big]\right\}$$

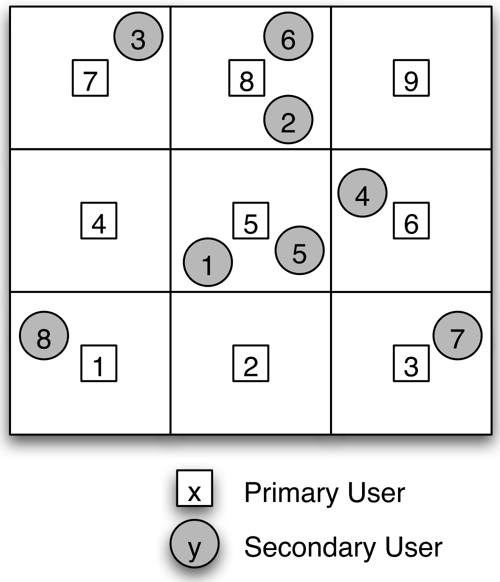

Figure 2.4: Example cell-partitioned network used in simulation

Using (2.29) and (2.30), we get the following:

$$\Delta^{GMM}(t) - V\mathbb{E}\left\{\sum_{n=1}^{N}\theta_n R_n^{GMM}(t)\right\} \leq B^{GMM} - \frac{V}{2}\sum_{n=1}^{N}\theta_n r_n^*$$

This is in a form that fits (2.19). Thus, applying Theorem 2 proves (2.32).

2.7 Simulations

We simulate the CNC algorithm on an example cognitive network consisting of 9 primary users and 8 secondary users as shown in Fig. 2.4. We consider a simple cell-partitioned network with one primary user per cell. The primary users are static and each has its own licensed channel that can be used by them simultaneously. A secondary user can only attempt to transmit on the channel associated with the primary user in its current cell.

The secondary users move from one cell to another according to a Markovian random walk. In particular, at the end of every slot, a secondary user decides to stay in its current cell with probability $1-\beta$, else decides to move to an adjacent cell with probability $\beta/4$ (where $\beta = 0.25$ for the simulations). If there is no feasible adjacent cell (e.g., if the previous cell is a corner cell and the new chosen cell does not exist), then the user remains in the current cell. It can be shown that the resulting $\boldsymbol{H}(t)$ process forms an irreducible, aperiodic Markov Chain where the steady state location distribution is uniform over all cells.

The channel state process $S_m(t)$ for each primary user m is governed by an ON/OFF Markov Chain with symmetric transition probabilities between the ON and OFF states given by 0.2 $\forall m$. The maximum collision fraction $\rho_m = 0.05\ \forall m$ so that for each primary user, at most 5% of its packets can have collisions.

New packets arrive at the secondary users according to independent Bernoulli processes, so that a single packet arrives i.i.d. with probability λ every slot. We assume there are no transport layer storage buffers, so that all packets that are not immediately admitted to the network layer are necessarily dropped. Admission control is performed according to (2.9) (with $\theta_n = 1\ \forall n$) and resource allocation decisions are made every slot according to (2.11). In this particular cell-partitioned network structure with one channel per cell, the maximum weight match can be decoupled into a distributed algorithm implemented in each cell, and is the same as the greedy maximal match that selects the largest weight user to transmit in each cell.

In Fig. 2.5 we plot the average total occupancy (summing all packets in the queues of the secondary users) versus the input rate λ . Each data point represents a simulation over

500, 000 timeslots, and the different curves correspond to values of the control parameter $V \in \{1, 2, 5, 10, 100\}$, and the case $V = \infty$ (no admission control) is also shown. In this case, the average total occupancy increases without bound as the input rate approaches network capacity. The vertical asymptote which appears roughly at $\lambda = 0.13$ packets/slot corresponds to this value. Fig. 2.6 illustrates the achieved throughput versus the raw data input rate λ for various V parameters. The achieved throughput is almost identical to the input rate λ for small values of λ, and the throughput saturates at a value that depends on V, being very close to the 0.13 capacity level when V is large.

Also, it was found that all real and virtual queue backlogs are always bounded by the maximum values given in (2.15) and (2.17). In particular, $\epsilon = 0.2$ for this network, so that $X_m(t) \leq X_{max} = Q_{max}\frac{1-\epsilon}{\epsilon} + 1 = 4Q_{max} + 1 = 4V + 5$. Finally, the maximum average fraction of collisions was very close to the target $\rho_m = 5\%$.

2.8 Chapter Summary

In this chapter, we developed an opportunistic scheduling algorithm for cognitive radio networks that maximizes the throughput utility of the secondary users subject to maximum collision constraints with the primary users. We used the recently developed technique of Lyapunov optimization along with the notion of collision queues to design an online admission control, scheduling and resource allocation algorithm. This algorithm provides tight reliability guarantees in terms of the worst case number of collisions suffered by a primary user in any time interval. Further, its performance can be pushed arbitrarily close to the optimal value with a trade-off in the average delay.

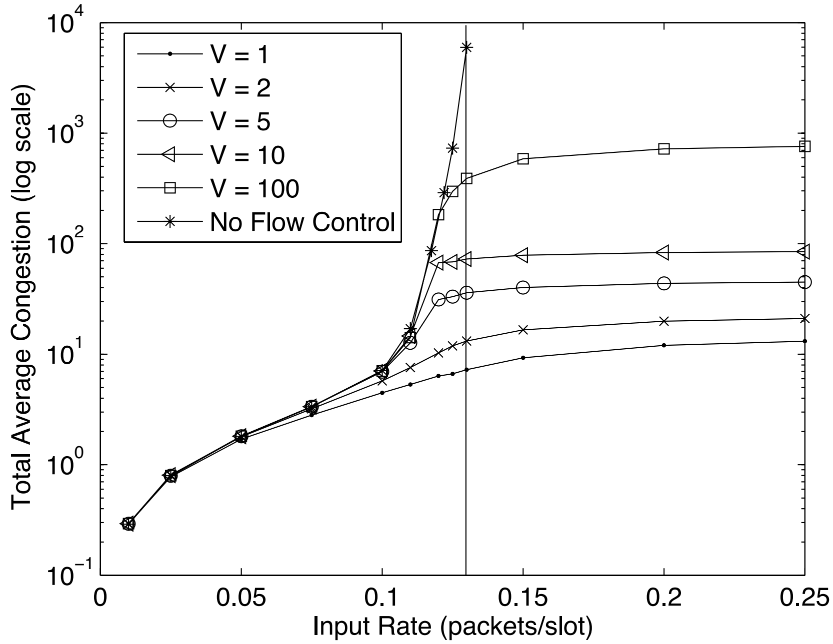

Figure 2.5: Total average congestion vs. input rate for different values of V

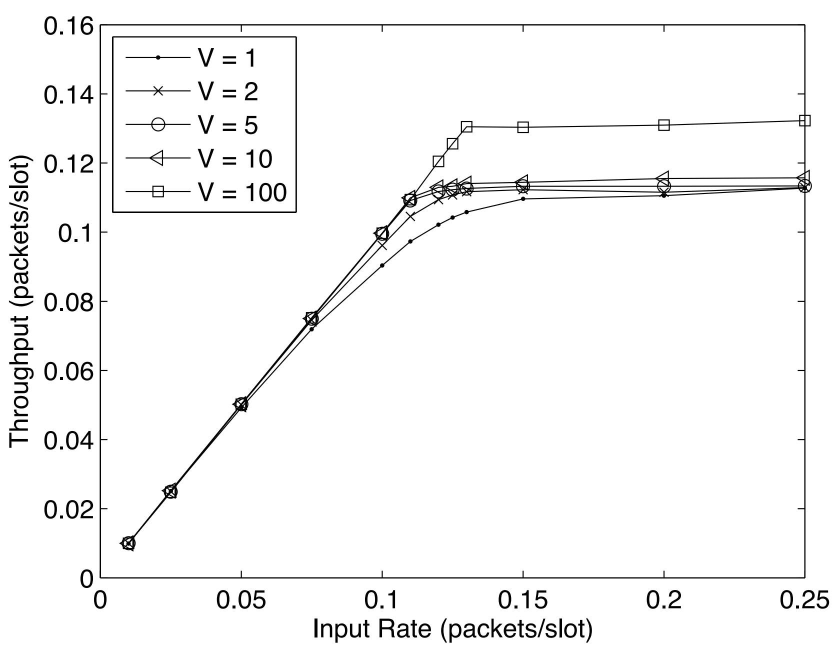

Figure 2.6: Achieved throughput vs. input rate for different values of V

Chapter 3

Delay-Limited Cooperative Communication

In this chapter, we investigate optimal resource allocation for *delay-limited* cooperative communication in time varying wireless networks. Specifically, we consider a team of mobile users with real-time applications that have strict delay constraints and fixed rate and reliability requirements (e.g., voice, multimedia). It is challenging to meet these requirements in such networks of power constrained devices, especially in the presence of mobility. Cooperative communication ("network MIMO") is a promising new physical layer technique to improve the performance of wireless networks. Cooperative communication protocols provide spatial diversity gains by making use of multiple relays for cooperative transmissions. This can be used to increase the reliability and/or reduce the energy costs of data transmissions. Cooperative communication is particularly attractive in such delay-limited scenarios since it can offer significant spatial diversity gains on top of conventional techniques used for combating fading.

Most prior work on cooperative communication has looked at physical layer resource allocation for a static network, particularly in the case of a single source. In mobile networks, the set of relay nodes varies over time. Further, the mobility patterns may be unknown to the network controller. Secondly, with multiple sources, the resources of the relays must be shared in a fair manner across all users in the network. Finally, the optimal strategy may involve a mixture of different modes of operation (direct transmission, multi-hop transmission and cooperative relaying) to meet the target reliability and average power constraints and the relaying modes must select different relay sets over time to achieve the optimal time average mixture. In this chapter, we overcome these challenges by designing a dynamic resource allocation algorithm that takes optimal control actions every slot and can be implemented in an online fashion. Using the tools of stochastic network optimization, we prove that our algorithm is guaranteed to achieve the target reliability and average power constraints whenever it is feasible to do so under any algorithm. Our algorithm can be used to significantly improve the performance of cooperative communication protocols in mobile ad-hoc networks with delay-limited traffic. We provide a general framework which can be applied to several cooperative protocols proposed in the literature (such as Amplify-and-Forward, Decode-and-Forward, etc.). Our work is the first to treat the problem of delay-limited cooperative communication with reliability constraints in a stochastic network characterized by fading channels, node mobility, and random packet arrivals, where opportunistic cooperation decisions are required.

3.1 Introduction

There is growing interest in the idea of utilizing cooperative communication [KMY06, SGL06, LTW04, LW03, SEA03a, SEA03b] to improve the performance of wireless networks with time varying channels. The motivation comes from the work on MIMO

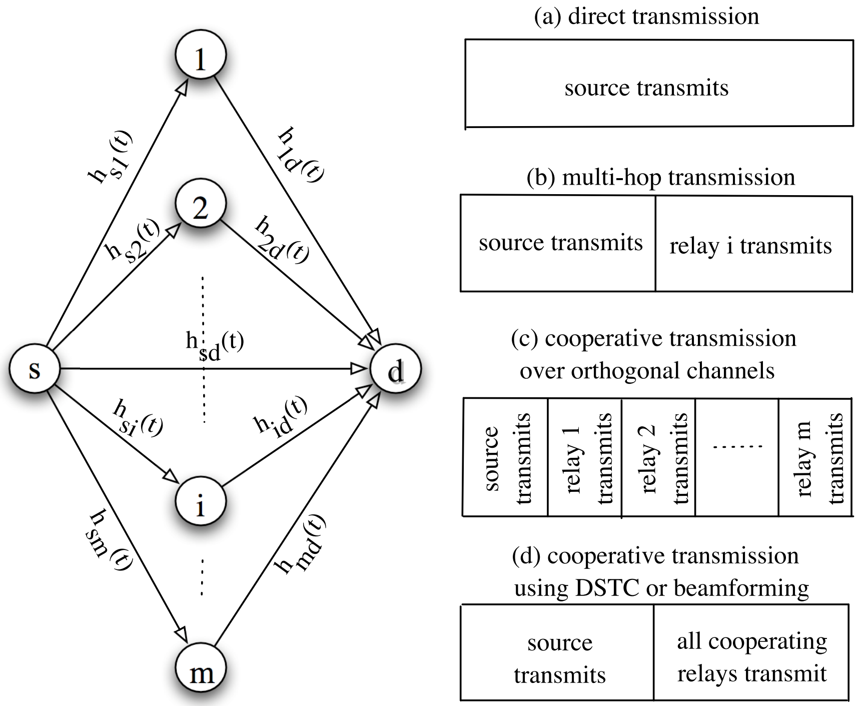

Figure 3.1: Example 2-hop network with source, destination and relays. The time slot structures for different transmission strategies are also shown. Due to the half-duplex constraint, cooperative protocols need to operate in two phases.

systems [TV05] which shows that employing multiple antennas on a wireless node can offer substantial benefits. However, this may be infeasible in small-sized devices due to space limitations. Cooperative communication has been proposed as a means to achieve the benefits of traditional MIMO systems using *distributed single antenna* nodes. Much recent work in this area promises significant gains in several metrics of interest (such as diversity [LTW04] [LW03], capacity [SEA03a, SEA03b, GV05, KGG05, HMZ05], energy efficiency [HA04, HHCK07], etc.) over conventional methods. We refer the interested reader to a recent comprehensive survey [KMY06] and its references.

The main idea behind cooperative communication can be understood by considering a simple 2-hop network consisting of a source s, its destination d and a set of m relay nodes as shown in Fig. 3.1. Suppose s has a packet to send to d in timeslot t. The channel gains for all links in this network are shown in the figure. In direct communication, s uses the full slot to transmit its packet to d over link $s - d$ as shown in Fig. 3.1(a). In conventional multi-hop relaying, s uses the first half of the slot to transmit its packet to a particular relay node i over link $s - i$ as shown in Fig. 3.1(b). If i can successfully decode the packet, it re-encodes and transmits it to d in the second half of the slot over link $i - d$. In both scenarios, to ensure reliable communication, the source and/or the relay must transmit at high power levels when the channel quality of any of the links involved is poor. However, note that due to the broadcast nature of wireless transmissions, other relay nodes may receive the signal from the transmission by s and can cooperatively relay it to d. The destination now receives multiple copies/signals and can use all of them jointly to decode the packet. Since these signals have been transmitted over independent paths, the probability that all of them have poor quality is significantly smaller. Cooperative communication protocols take advantage of this *spatial diversity gain* by making use of multiple relays for cooperative transmissions to increase reliability and/or reduce energy costs. This is different from traditional multi-hop relaying in which only one node is responsible for forwarding at any time and in which the destination does not use multiple signals to decode a packet.

Because of the half-duplex nature of wireless devices, a relay node cannot send and receive on the same channel simultaneously. Therefore, such cooperative communication protocols typically operate over a two phase slot structure as shown in Figs. 3.1(c) and

3.1(d). In the first phase, s transmits its packet to the set of relay nodes. In the second phase, a subset of these relays transmit their signals to d. Note that the destination may receive the source signal from the first phase as well. At the end of the second phase, the destination appropriately combines all of these received signals to decode the packet. The exact slot structure as well as the signals transmitted by the relays depend on the cooperative protocol being used.[1] For example, Fig. 3.1(c) shows the slot structure under a cooperative scheme that transmits over orthogonal channels. Specifically, the time slot is divided into $m+1$ equal mini-slots. In phase one, the source transmits its packet in the first mini-slot. In the second phase, the relays transmit one after the other in their own mini-slots. Fig. 3.1(d) shows the slot structure under a cooperative scheme in which the cooperating relays use distributed space-time codes (DSTC) or a beamforming technique to transmit simultaneously in the second phase. It should be noted that due to this half-duplex constraint, there is an inherent loss in the multiplexing gain under any such cooperative transmission strategy over direct transmission. Therefore, it is important to develop algorithms that cooperate opportunistically.

In this work, we consider a mobile ad-hoc network with *delay-limited* traffic and cooperative communication. Many real-time applications (e.g., voice) have stringent delay constraints and fixed rate requirements. In slow fading environments (where decoding delay is of the order of the channel coherence time), it may not be possible to meet these delay constraints for every packet. However, these applications can often tolerate a certain fraction of lost packets or outages. A variety of techniques are used to combat fading and meet this target outage probability (including exploiting diversity, channel coding, ARQ,

[1] We consider several protocol examples in Sec. 3.5

power control, etc.). Cooperative communication is a particularly attractive technique to improve reliability in such delay-limited scenarios since it can offer significant spatial diversity gains in addition to these techniques.

Much prior work on cooperative communication considers physical layer resource allocation for a static network, particularly in the case of a single source. Objectives such as minimizing sum power, minimizing outage probability, meeting a target SNR constraint, etc., are treated in this context [HMZ05, HA04, HHCK07, MY04b, MY10, ZAL07, GE07, CSY08]. We draw on this work in the development of *dynamic* resource allocation in a stochastic network with fading channels, node mobility, and random packet arrivals, where *opportunistic cooperation decisions* are required. Dynamic cooperation was also considered in the prior work [YB07] which investigates throughput optimality and queue stability in a multi-user network with static channels and randomly arriving traffic using the framework of Lyapunov drift. Our formulation is different and does not involve issues of queue stability. Rather, we consider a delay-limited scenario where each packet must either be transmitted in one slot, or dropped. This is similar to the concept of *delay-limited capacity* [HT98]. Also related to such scenarios is the notion of *minimum outage probability* [CTB99]. These quantities are also investigated in the recent work [GE07] that considers a 3 node static network with Rayleigh fading and shows that opportunistic cooperation significantly improves the delay-limited capacity.

In this work, we use techniques of both Lyapunov drift and Lyapunov optimization [GNT06] to develop a control algorithm that takes dynamic decisions for each new slot. Different from most work that applies this theory, our solution involves a 2-stage stochastic shortest path problem due to the cooperative relaying structure. This problem is non-convex and combinatorial in nature and does not admit closed form solutions in general. However, under several important and well known classes of physical layer co-operation models, we develop techniques for reducing the problem exactly to an m-stage set of convex programs. The convex programs themselves are shown to have quasi-closed form solutions and can be computed in real time for each slot, often involving simple water-filling strategies that also arise in related static optimization problems.

3.2 Basic Network Model

We consider a mobile ad-hoc network with delay-limited communication over time varying fading channels. The network contains a set $\mathcal{N}$ of nodes, all potentially mobile. All nodes are assumed to be within range of each other, and any node pair can communicate either through direct transmission or through a 2-phase cooperative transmission that makes use of other nodes as relays. The system operates in slotted time and the channel coefficient between nodes i and j in slot t is denoted by $h_{ij}(t)$. We assume a block fading model [TV05] for the channel coefficients so that their value remains fixed during a slot and changes from one slot to the other according to the distribution of the underlying fading and mobility processes.

For simplicity, we assume that the set $\mathcal{N}$ contains a single source node s and its destination node d and that all other nodes act simply as cooperative relays. This is similar to the single-source assumption treated in [MY04b, MY10, GE07, CSY08, ZAL07] for static networks. We derive a dynamic cooperation strategy for this single source problem in Sec. 3.4 that optimizes a weighted sum of reliability and power expenditure subject to individual reliability and average power constraints at the source and at all relays. This highlights the decisions involved from the perspective of a source node, and these decisions and the resulting solution structure are similar to the multi-source scenario operating under an orthogonal medium access scheme (such as TDMA or FDMA) studied later in Sec. 3.7. In the following, we denote the set of relay nodes by $\mathcal{R}$ and the set $\{s\} \cup \mathcal{R}$ by $\widehat{\mathcal{R}}$. All nodes $i \in \widehat{\mathcal{R}}$ have both long term average and instantaneous peak power constraints given by P_i^{avg} and P_i^{max} respectively.

We consider two models for the availability of the channel state information (CSI). The first is the *known channels, unknown statistics* model. Under this model, we assume that the channel gains between the source node and its relay set and destination as well as the channel gains between the relays and the destination are known every slot. These could be obtained by sending pilot signals and via feedback. This model has also been considered in prior works [MY04b,MY10,GE07,CSY08] on power allocation in static networks where, in addition to the current channel gains, a knowledge of the distribution governing the fading process is assumed. In our work, under this *known channels, unknown statistics* model, we do not assume any knowledge of the distributions governing the evolution of the channel states, mobility processes, or traffic. Thus, our algorithm and its optimality properties hold for a very general class of channel and mobility models that satisfy certain ergodicity requirements (to be made precise later). We note that the channel gain could represent just the amplitude of the channel coefficient if an orthogonal cooperative scheme is being used. However, in case of cooperative schemes such as beamforming, this could represent the complete description of the fading coefficient that includes the phase information.

The second model we consider is the *unknown channels, known statistics* model. In this case, we assume that the current set of potential relay nodes is known on each slot t, but the exact channel realizations between the source and these relays, and the relays and the destination, are unknown. Rather, we assume only that the *statistics* of the fading coefficients are known between the source and current relays, and the current relays and destination. However, we still do not require knowledge of the distributions governing the arriving traffic or the mobility pattern (which affects the set of relays we will see in future slots). This is in contrast to prior works that have considered resource allocation in the presence of partial CSI only for static networks.

For both models, we use $\mathcal{I}(t)$ to represent the collection of all channel state information known on slot t. For the known channels, unknown statistics model, $\mathcal{I}(t)$ represents the collection of channel coefficients $h_{ij}(t)$ between the source and relays and relays and destination. For the unknown channels, known statistics model, $\mathcal{I}(t)$ represents the set of all nodes that are available on slot t for relaying and the distribution of the fading coefficients. We assume that $\mathcal{I}(t)$ lies in a space of finite but arbitrarily large size and evolves according to an ergodic process with a well defined steady state distribution. This variation in channel state information affects the reliability and power expenditure associated with the direct and cooperative transmission modes that are discussed in Sec. 3.2.2.

3.2.1 Example of Channel State Information Models

As an example of these models, suppose the nodes move in a cell-partitioned network according to a Markovian random walk (see also Fig. 3.2 in Sec. 3.8 on Simulations).

Each slot, a node may decide to stay in its current cell or move to an adjacent cell according to the probability distribution governing the random walk. Suppose that each slot, the set of potential relays consists only of nodes in either the same or an adjacent cell of the source. Suppose channel gains between nodes in the same cell are distributed according to a Rayleigh fading model with a particular mean and variance, while gains for nodes in adjacent cells are Rayleigh with a different mean and variance. Under the known channels, unknown statistics model, the $\mathcal{T}(t)$ information is the set of current gains $h_{ij}(t)$, and the Rayleigh distribution is not needed. Under the unknown channels, known statistics model, the $\mathcal{T}(t)$ information is the set of nodes currently in the same and adjacent cells of the source, and we assume we know that the fading distribution is Rayleigh, and we know the corresponding means and variances. However, neither model requires knowledge of the mobility model or the traffic rates.

3.2.2 Control Options

Suppose the slot size is normalized to integer slots $t \in \{0, 1, 2, \ldots, \}$. In each slot, the source s receives new packets for its destination d according to an i.i.d. Bernoulli process $A_s(t)$ of rate λ_s. Each packet is assumed to be R bits long and has a *strict* delay constraint of 1 slot. Thus, a packet not served within 1 slot of its arrival is dropped. Further, packets that are not successfully received by their destinations due to channel errors are not retransmitted. The source node has a minimum time-average reliability requirement specified by a fraction ρ_s which denotes the fraction of packets that were transmitted successfully. In any slot t, if source s has a new packet for transmission, it can use one of the following transmission modes (Fig. 3.1):

1. Transmit directly to d using the full slot

2. Transmit to d using traditional relaying over two hops

3. Transmit cooperatively with the set $\mathcal{R}$ of relay nodes using the two phase slot structure

4. Stay idle (so that the packet gets dropped)

We consider all of these transmission modes because, depending on the current channel conditions and energy costs in slot t, it might be better to choose one over the other. For example, due to the half-duplex constraint, direct transmission using the full slot might be preferable to cooperative transmission over two phases on slots when the source-destination link quality is good. Note that this is similar to the much studied framework of opportunistic transmission scheduling in time varying channels. Further, even in the special case of static channels, the optimal strategy may involve a mixture of these modes of operation to meet the target reliability and average power constraints.

Let $\mathcal{I}^{\eta}(t)$ denote the collective control action in slot t under some policy η that includes the choice of the transmission mode at the source, power allocations for the source and all relevant relays, and any additional physical layer choices such as modulation and coding. Specifically, we have $\mathcal{I}^{\eta}(t) = [\text{mode choice}, \boldsymbol{P}^{\eta}(t), \text{other PHY layer choices}]$ where the mode choice refers to one of the 4 transmission modes for the source, and where $\boldsymbol{P}^{\eta}(t)$ is the collection of coefficients $P_i^{\eta}(t)$ representing power allocations for each node $i \in \widehat{\mathcal{R}}$. Note that $P_i^{\eta}(t) = 0$ for all i under transmission mode 4 (idle). If the source s chooses mode 1, we have $P_i(t) = 0$ for all relay nodes $i \in \mathcal{R}$, whereas if s chooses mode 2, we have $P_i(t) > 0$ for at most one relay $i \in \mathcal{R}$. Note that under any feasible policy η, $P_i^{\eta}(t)$ must satisfy the instantaneous peak power constraint every slot for all i. Also note that under the cooperative transmission option, the power allocation for the source node and the relays corresponds to the first and second phase respectively. Thus, the source is active in the first phase while the relays are active in the second phase. We denote the set of all valid power allocations by $\mathcal{P}$ and define $\mathcal{C}$ as the set of all valid control actions:

$$\mathcal{C} = \{1, 2, 3, 4\} \times \{\mathcal{P}\} \times \{\text{other PHY layer choices}\}$$

The success/failure outcome of the control action is represented by an indicator random variable $\Phi_s(\mathcal{I}^\eta(t), \mathcal{T}(t))$ that depends on the current control action and channel state. Successful transmission of a packet is usually a complicated function of the transmission mode chosen, the associated power allocations and channel states, as well as physical layer details like modulation, coding/decoding scheme, etc. In this work, the particular physical layer actions are included in the $\mathcal{I}^\eta(t)$ decision variable. Specifically, given a control action $\mathcal{I}^\eta(t)$ and a channel state $\mathcal{T}(t)$, the outcome is defined as follows:

$$\Phi_s(\mathcal{I}^\eta(t), \mathcal{T}(t)) \triangleq \begin{cases} 1 & \text{if a packet transmitted by } s \text{ in slot } t \text{ is successfully received by } d \\ 0 & \text{else} \end{cases} \tag{3.1}$$

Note that $\Phi_s(\mathcal{I}^\eta(t), \mathcal{T}(t))$ is a random variable, and its conditional expectation given $(\mathcal{I}^\eta(t), \mathcal{T}(t))$ is equal to the success probability under the given physical layer channel model. Use of this abstract indicator variable allows a unified treatment that can include a variety of physical layer models. Under the known channels, unknown statistics model (where $\mathcal{T}(t)$ includes the full channel realizations between source and relays and relays and destination on slot t), $\Phi_s(\mathcal{I}^\eta(t), \mathcal{T}(t))$ can be a deterministic 0/1 function based on the known channel state and control action. Specific examples for this model are considered in Sec. 3.5. Under the unknown channels, known statistics model (where $\mathcal{T}(t)$ represents only the set of current possible relays and the fading statistics), we assume we know the value of $Pr[\Phi_s(\mathcal{I}^\eta(t), \mathcal{T}(t)) = 1]$ under each possible control action $\mathcal{I}^\eta(t)$. This model is considered in Sec. 3.6. Under both models, we assume that explicit ACK/NACK information is received at the end of each slot, so that the source knows the value of $\Phi_s(\mathcal{I}^\eta(t), \mathcal{T}(t))$. For notational convenience, in the rest of the chapter, we use $\Phi_s^\eta(t)$ instead of $\Phi_s(\mathcal{I}^\eta(t), \mathcal{T}(t))$ noting that the dependence on $(\mathcal{I}^\eta(t), \mathcal{T}(t))$ is implicit.

3.2.3 Discussion of Basic Model

The basic model described above extends prior work on 2-phase cooperation in static networks to a mobile environment, and treats the important example scenario where a team of nodes move in a tight cluster but with possible variation in the relative locations of nodes within the cluster. We note that our model and results are applicable to the special case of a static network as well. Another example scenario captured by our model is an OFDMA-based cellular network with multiple users that have both inter-cell and intra-cell mobility. In each slot, a set of transmitters is determined in each orthogonal channel (for example, based on a predetermined TDMA schedule, or dynamically chosen by the base station). The remaining nodes can potentially act as cooperative relays in that slot.

The basic model treats scenarios in which a source node can transmit to its destination, possibly with the help of multiple relay nodes, in 2 stages. While this is a simplifying assumption, the framework developed here can be applied to more general scenarios in which, in a single slot, cooperative relaying over K stages is performed (for some $K > 2$) using multi-hop cooperative techniques (e.g., [SMSM06, BZG07]).

3.3 Control Objective

Let α_s and β_i for $i \in \widehat{\mathcal{R}}$ be a collection of non-negative weights. Then our objective is to design a policy η that solves the following *stochastic optimization problem*:

$$\begin{aligned}
\text{Maximize:} \quad & \alpha_s \bar{r}_s^\eta - \sum_{i \in \widehat{\mathcal{R}}} \beta_i \bar{e}_i^\eta \\
\text{Subject to:} \quad & \bar{r}_s^\eta \geq \rho_s \lambda_s \\
& \bar{e}_i^\eta \leq P_i^{avg} \; \forall \, i \in \widehat{\mathcal{R}} \\
& 0 \leq P_i^\eta(t) \leq P_i^{max} \; \forall \, i \in \widehat{\mathcal{R}}, \; \forall t \\
& \mathcal{I}^\eta(t) \in \mathcal{C} \; \forall t
\end{aligned} \tag{3.2}$$

where $\bar{r}_s^\eta$ is the time average reliability for source s under policy η and is defined as:

$$\bar{r}_s^\eta \triangleq \lim_{t \to \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}\left\{\Phi_s^\eta(\tau)\right\} \tag{3.3}$$

and $\bar{e}_i^\eta$ is the time average power usage of node i under η:

$$\bar{e}_i^{\eta} \triangleq \lim_{t\to\infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}\left\{P_i^{\eta}(\tau)\right\} \tag{3.4}$$

Here, the expectation is with respect to the possibly randomized control actions that policy η might take. The α_s and β_i weights allow us to consider several different objectives. For example, setting $\alpha_s = 0$ and $\beta_i = 1$ for all i reduces (3.2) to the problem of minimizing the average sum power expenditure subject to minimum reliability and average power constraints. This objective can be important in the multiple source scenario when the resources of the relays must be shared across many users. Setting all of these weights to 0 reduces (3.2) to a feasibility problem where the objective is to provide minimum reliability guarantees subject to average power constraints.

Problem (3.2) is similar to the general stochastic utility maximization problem presented in [GNT06]. Suppose (3.2) is feasible and let r_s^* and e_i^* $\forall i \in \widehat{\mathcal{R}}$ denote the optimal value of the objective function, potentially achieved by some arbitrary policy. Using the techniques developed in [GNT06, Nee06], it can be shown that it is sufficient to consider only the class of stationary, randomized policies that take control decisions purely as a (possibly random) function of the channel state $\mathcal{T}(t)$ every slot to solve (3.2). However, computing the optimal stationary, randomized policy explicitly can be challenging and often impractical as it requires knowledge of arrival distributions, channel probabilities and mobility patterns in advance. Further, as pointed out earlier, even in the special case of a static channel, the optimal strategy may involve a mixture of direct transmission, multi-hop, and cooperative modes of operation, and the relaying modes must select different relay sets over time to achieve the optimal time average mixture.

However, the technique of Lyapunov optimization [GNT06] can be used to construct an alternate dynamic policy that overcomes these challenges and is provably optimal. Unlike the stationary, randomized policy, this policy does not need to be computed beforehand and can be implemented in an online fashion. In the known channels model, it does not need a-priori statistics of the traffic, channels, or mobility. In the unknown channels model, it does not need a-priori statistics of the traffic or mobility. We present this policy in the next section.

3.4 Optimal Control Algorithm

In this section, we present a dynamic control algorithm that achieves the optimal solution r_s^* and e_i^* $\forall i \in \widehat{\mathcal{R}}$ to the stochastic optimization problem presented earlier. This algorithm is similar in spirit to the backpressure algorithms proposed in [GNT06,Nee06] for problems of throughput and energy optimal networking in time varying wireless ad-hoc networks.

The algorithm makes use of a "reliability queue" $Z_s(t)$ for source s. Specifically, let $Z_s(t)$ be a value that is initialized to zero (so that $Z_s(0) = 0$), and that is updated at the end of every slot t according to the following equation:

$$Z_s(t+1) = \max[Z_s(t) - \Phi_s(t), 0] + \rho_s A_s(t) \tag{3.5}$$

where $A_s(t)$ is the number of arrivals to source s on slot t (being either 0 or 1), and $\Phi_s(t)$ is 1 if and only if a packet that arrived was successfully delivered (recall that ACK/NACK information gives the value of $\Phi_s(t)$ at the end of every slot t). Additionally, it also uses the following virtual power queues $\forall i \in \widehat{\mathcal{R}}$:

$$X_i(t+1) = \max[X_i(t) - P_i^{avg}, 0] + P_i(t) \tag{3.6}$$

All these queues are also initialized to 0 and updated at the end of every slot t according to the equation above. We note that these queues are virtual in that they do not represent any real backlog of data packets. Rather, they facilitate the control algorithm in achieving the time average reliability and energy constraints of (3.2) as follows. If a policy η stabilizes (3.5), then we must have that its service rate is no smaller than the input rate, i.e.,

$$\bar{r}_s^\eta = \lim_{t\to\infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}\left\{\Phi_s^\eta(\tau)\right\} \geq \lim_{t\to\infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}\left\{\rho_s A_s(\tau)\right\} = \rho_s \lambda_s$$

Similarly, stabilizing (3.6) yields the following:

$$\bar{e}_i^\eta = \lim_{t\to\infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}\left\{P_i^\eta(\tau)\right\} \leq P_i^{avg}$$

where we have used definitions (3.3), (3.4). This technique of turning time-average constraints into queueing stability problems was first used in [Nee06].

To stabilize these virtual queues and optimize the objective function in (3.2), the algorithm operates as follows. Let $\boldsymbol{Q}(t) = (Z_s(t), X_i(t))\ \forall i \in \widehat{\mathcal{R}}$ denote the collection of these queues in timeslot t. Every slot t, given $\boldsymbol{Q}(t)$ and the current channel state $\mathcal{T}(t)$, it chooses a control action $\mathcal{I}^*(t)$ that minimizes the following stochastic metric (for a given control parameter $V \geq 0$):

$$
\begin{aligned}
\text{Minimize:} \quad & (X_s(t) + V\beta_s)\mathbb{E}\left\{P_s(t)|\boldsymbol{Q}(t), \mathcal{T}(t)\right\} + \sum_{i\in\mathcal{R}}(X_i(t) + V\beta_i)\mathbb{E}\left\{P_i(t)|\boldsymbol{Q}(t), \mathcal{T}(t)\right\} \\
& - (Z_s(t) + V\alpha_s)\mathbb{E}\left\{\Phi_s(t)|\boldsymbol{Q}(t), \mathcal{T}(t)\right\} \\
\text{Subject to:} \quad & 0 \leq P_i(t) \leq P_i^{max} \; \forall i \in \widehat{\mathcal{R}} \\
& \mathcal{I}(t) \in \mathcal{C} \qquad (3.7)
\end{aligned}
$$

After implementing $\mathcal{I}^*(t)$ and observing the outcome, the virtual queues are updated using (3.5), (3.6). Recall that there are no actual queues in the system. Our algorithm enforces a strict 1-slot delay constraint so that $\Phi_s(t) = 0$ if the packet is not successfully delivered after 1 slot. The virtual queues $X_i(t), Z_s(t)$ are maintained only in software and act as known weights in the optimization (3.7) that guide decisions towards achieving our time average power and reliability goals. The control action $\mathcal{I}^*(t)$ that optimizes (3.7) affects the powers $P_i(t)$ allocated and the $\Phi_s(t)$ value according to (3.1).

The above optimization is a 2-stage *stochastic shortest path* problem [Ber07] where the two stages correspond to the two phases of the underlying cooperative protocol. Specifically, when s decides to use the option of transmitting cooperatively, the cost incurred in the first stage is given by the first term $(X_s(t)+V\beta_s)\mathbb{E}\left\{P_s(t)|\boldsymbol{Q}(t), \mathcal{T}(t)\right\}$. The cost incurred during the second stage is given by $\sum_{i\in\mathcal{R}}(X_i(t) + V\beta_i)\mathbb{E}\left\{P_i(t)|\boldsymbol{Q}(t), \mathcal{T}(t)\right\}$ and at the end of this stage, we get a reward of $(Z_s(t) + V\alpha_s)\mathbb{E}\left\{\Phi_s(t)|\boldsymbol{Q}(t), \mathcal{T}(t)\right\}$. The transmission outcome $\Phi_s(t)$ depends on the power allocation decisions in *both* phases which makes this problem different from greedy strategies (e.g., [YB07], [Nee06]). In order to determine the optimal strategy in slot t, the source s computes the minimum cost of (3.7) for all transmission modes described earlier and chooses one with the least cost.

Note that this problem is unconstrained since the long term time average reliability and power constraints do not appear explicitly as in the original problem. These are implicitly captured by the virtual queue values. Further, its solution uses the value of the *current* channel state $\mathcal{T}(t)$ and does not require knowledge of the statistics that govern the evolution of the channel state process. Thus, the control strategy involves implementing the solution to the sequence of such unconstrained problems every slot and updating the queue values according to (3.5), (3.6). Assuming i.i.d. $\mathcal{T}(t)$ states, the following theorem characterizes the performance of this dynamic control algorithm A similar statement can be made for more general Markov modulated $\mathcal{T}(t)$ using the techniques of [GNT06]. For simplicity, here we consider the i.i.d. case.

Theorem 4 *(Algorithm Performance) Suppose all queues are initialized to* 0*. Then, implementing the dynamic algorithm (3.7) every slot stabilizes all queues, thereby satisfying* *the minimum reliability and time-average power constraints, and guarantees the following performance bounds (for some $\epsilon > 0$ that depends on the slackness of the feasibility constraints):*

$$\lim_{t\to\infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}\left\{Z_s(\tau)\right\} \leq \frac{B + V(\alpha_s + \sum_{i\in\widehat{\mathcal{R}}} \beta_i P_i^{max})}{\epsilon} \tag{3.8}$$

$$\lim_{t\to\infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \sum_{i\in\widehat{\mathcal{R}}} \mathbb{E}\left\{X_i(\tau)\right\} \leq \frac{B + V(\alpha_s + \sum_{i\in\widehat{\mathcal{R}}} \beta_i P_i^{max})}{\epsilon} \tag{3.9}$$

Further, the time average utility achieved for any $V \geq 0$ satisfies:

$$\lim_{t\to\infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}\left\{\alpha_s \Phi_s(\tau) - \sum_{i\in\widehat{\mathcal{R}}} \beta_i P_i(\tau)\right\} \geq \zeta^* - \frac{B}{V} \tag{3.10}$$

where

$$\zeta^* \triangleq \alpha_s r_s^* - \sum_{i\in\widehat{\mathcal{R}}} \beta_i e_i^* \tag{3.11}$$

$$B \triangleq \frac{1 + \lambda_s^2 \rho_s^2 + \sum_{i\in\widehat{\mathcal{R}}} (P_i^{avg})^2 + (P_i^{max})^2}{2} \tag{3.12}$$

Proof 4 *See Appendix B.1.*

Thus, one can get within $O(1/V)$ of the optimal values by increasing V at the cost of an $O(V)$ increase in the virtual queue backlogs. The size of these queues affects the time required for the time average values to converge to the desired performance.

In the following sections, we investigate the basic 2-stage resource allocation problem (3.7) in detail and present solutions for two widely studied classes of cooperative protocols proposed in the literature: Decode-and-Forward (DF) and Amplify-and-Forward (AF) [LTW04, LW03]. These protocols differ in the way the transmitted signal from the first phase is processed by the cooperating relays. In DF, a relay fully decodes the signal. If the packet is received correctly, it is re-encoded and transmitted in the second phase. In AF, a relay simply retransmits a scaled version of the received analog signal. We refer to [LTW04, LW03] for further details on the working of these protocols as well as derivation of expressions for the mutual information achieved by them. Let $m = |\mathcal{R}|$. In the following, we assume a Gaussian channel model with a total bandwidth W and unit noise power per dimension. We use the information theoretic definition of a transmission failure (an outage event) as discussed in [HT98], [CTB99]. Here, an outage occurs when the total instantaneous mutual information is smaller than the rate R at which data is being transmitted.

We first consider the case when the channel gains are known at the source (Sec. 3.5). In this scenario, (3.7) becomes a 2-stage *deterministic shortest path problem* because the outcome $\Phi_s(t)$ due to any control decision and its power allocation can be computed beforehand. Specifically, $\Phi_s(t) = 1$ when the resulting total mutual information exceeds R and $\Phi_s(t) = 0$ otherwise. Further, this outcome is a function of control actions taken over two stages when cooperative transmission is used. This resulting problem is combinatorial and non-convex and does not admit closed-form solutions in general. However, for these protocols, we can reduce it to a set of simpler convex programs for which we can derive quasi-closed form solutions. Then in Sec. 3.6, we consider the case when only the statistics of the channel gains are known. In this case, the outcome $\Phi_s(t)$ is random function of the control actions (taken over the two stages in case of cooperative transmission) and (3.7) becomes a 2-stage *stochastic dynamic program.* While standard dynamic programming techniques can be used to compute the optimal solution, they are typically computationally intensive. Therefore, for this case, we present a Monte Carlo simulation based technique to efficiently solve the resulting dynamic program.

3.5 Known Channels, Unknown Statistics

Recall that in order to determine the optimal control action in any slot t, we must choose between the four modes of operation as discussed in Sec. 3.2: (1) direct transmission, (2) multi-hop relay, (3) cooperative, and (4) idle. Let $c_i(t)$ and $I_i(t)$ denote the optimal cost of the metric (3.7), and the corresponding action that achieves that metric, assuming that mode $i \in \{1, 2, 3, 4\}$ is chosen in slot t. Every slot, the algorithm computes $c_i(t)$ and $I_i(t)$ for each mode and then implements the mode i and the resulting action $I_i(t)$ that minimizes cost. Note that the cost $c_4(t)$ for the idle mode is trivially 0. The minimum cost for direct transmission can be computed as follows. When the source transmits directly, we have $P_i(t) = 0 \; \forall i \in \mathcal{R}$. The minimum cost $c_1(t)$ associated with a *successful* direct transmission ($\Phi_s(t) = 1$) can be obtained by solving the following convex problem[2]:

[2]Note that the term $-Z_s(t) - V\alpha_s$ in the objective is a constant in any given slot and does not affect the solution. However, we keep it to compare the net cost between all modes of operation.

$$\begin{aligned}
\text{Minimize:} \quad & \Big(X_s(t) + V\beta_s\Big)P_s(t) - Z_s(t) - V\alpha_s \\
\text{Subject to:} \quad & W \log\Big(1 + \frac{P_s(t)}{W}|h_{sd}(t)|^2\Big) \geq R \\
& 0 \leq P_s(t) \leq P_s^{max} \qquad (3.13)
\end{aligned}$$

where the constraint $W \log\Big(1+\frac{P_s(t)}{W}|h_{sd}(t)|^2\Big) \geq R$ represents the fact that to get $\Phi_s(t) = 1$, the mutual information must exceed R. It is easy to see that if there is a feasible solution to the above, then for minimum cost, this constraint must be met with equality. Using this, the minimum cost corresponding to the direct transmission mode is given by: $\Big(X_s(t) + V\beta_s\Big)P_s^{dir}(t) - Z_s(t) - V\alpha_s$ if $P_s^{dir}(t) = \frac{W}{|h_{sd}(t)|^2}(2^{R/W} - 1) \leq P_s^{max}$. Otherwise, direct transmission is infeasible and so we set $c_1(t) = +\infty$. In this case, direct transmission will not be considered as the idle mode cost $c_4(t) = 0$ is strictly better, but we must also compare with the costs $c_2(t)$ and $c_3(t)$.

To compute the minimum cost $c_2(t)$ associated with multi-hop transmission, note that in this case, the slot is divided into two parts (Fig. 3.1(b)) and $P_i(t) > 0$ for at most one $i \in \mathcal{R}$. This strategy is a special case of the Regenerative DF protocol (to be discussed next) that uses only 1 relay and in which the destination does not use signals received from the first stage for decoding. Therefore, the optimal cost for this can be calculated using the procedure for the Regenerative DF case by imposing the single relay constraint and setting $h_{sd}(t) = 0$.

Below we present the computation of the minimum cost $c_3(t)$ for the cooperative transmission mode under several protocols. In what follows, we drop the time subscript (t) for notational convenience.

3.5.1 Regenerative DF, Orthogonal Channels

Here, the source and relays are each assigned an orthogonal channel of equal size. An example slot structure is shown in Fig. 3.1(c) in which the entire slot is divided into $m+1$ equal mini-slots. In the first phase of the protocol, s transmits the packet in its slot using power P_s. In the second phase, a subset $\mathcal{U} \subset \mathcal{R}$ of relays that were successful in reliably decoding the packet, re-encode it using the *same* code book and transmit to the destination on their channels with power P_i (where $i \in \mathcal{U}$). Given such a set $\mathcal{U}$, the total mutual information under this protocol is given by [LTW04]:

$$\frac{W}{m} \log\Big(1 + \frac{mP_s}{W}|h_{sd}|^2 + \sum_{i \in \mathcal{U}} \frac{mP_i}{W}|h_{id}|^2\Big)$$

This is derived by assuming that the receiver uses Maximal Ratio Combining to process the signals. As seen in the expression for the mutual information, such an orthogonal structure increases the SNR, but utilizes only a fraction of the available degrees of freedom leading to reduced multiplexing gain.

Define binary variables x_i to be 1 if relay i can reliably decode the packet after the first stage and 0 else. Then, for this protocol, (3.7) is equivalent to the following optimization problem:

$$\begin{aligned}
\text{Minimize:} \quad & (X_s + V\beta_s)P_s + \sum_{i\in\mathcal{R}} (X_i + V\beta_i)P_i - Z_s - V\alpha_s \\
\text{Subject to:} \quad & \frac{W}{m}\log\Big(1 + \frac{mP_s}{W}|h_{sd}|^2 + \sum_{i\in\mathcal{R}} x_i \frac{mP_i}{W}|h_{id}|^2\Big) \geq R \\
& \frac{W}{m}\log\Big(1 + \frac{mP_s}{W}|h_{si}|^2\Big) \geq x_i R \\
& 0 \leq P_s \leq P_s^{max} \\
& 0 \leq P_i \leq P_i^{max}, x_i \in \{0,1\}\ \forall i \in \mathcal{R}
\end{aligned} \tag{3.14}$$

The variables x_i capture the requirement that a relay can cooperatively transmit in the second stage only if it was successful in reliably decoding the packet using the first stage transmission. A similar setup is considered in [MY04b] but it treats the limiting case when W goes to infinity. Because of the integer constraints on x_i, (3.14) is non-convex. However, we can exploit the structure of this protocol to reduce the above to a set of $m+1$ subproblems as follows. We first order the relays in decreasing order of their $|h_{si}|^2$ values. Define $\mathcal{U}_k$ as the set that contains the first k (where $0 \leq k \leq m$) relays from this ordering. Let $P_s^{\mathcal{U}_k}$ denote the minimum source power required to ensure that all relays in $\mathcal{U}_k$ can reliably decode the packet after the first stage. We note that for all values of P_s in the range $(P_s^{\mathcal{U}_k}, P_s^{\mathcal{U}_{k+1}})$, the relay set that can reliably decode remains the same, i.e., $\mathcal{U}_k$. Thus, we need to consider only $m+1$ subproblems, one for each $\mathcal{U}_k$. The subproblem for any set $\mathcal{U}_k$ is given by:

$$\begin{aligned}
\text{Minimize:} \quad & (X_s + V\beta_s)P_s + \sum_{i\in\mathcal{U}_k}(X_i + V\beta_i)P_i - Z_s - V\alpha_s \\
\text{Subject to:} \quad & \frac{W}{m}\log\Big(1 + \frac{mP_s}{W}|h_{sd}|^2 + \sum_{i\in\mathcal{U}_k}\frac{mP_i}{W}|h_{id}|^2\Big) \geq R \\
& P_s^{\mathcal{U}_k} \leq P_s \leq P_s^{max} \\
& 0 \leq P_i \leq P_i^{max} \qquad \forall i \in \mathcal{U}_k
\end{aligned} \tag{3.15}$$

This can easily be expressed as the following LP:

$$\begin{aligned}
\text{Minimize:} \quad & (X_s + V\beta_s)P_s + \sum_{i\in\mathcal{U}_k}(X_i + V\beta_i)P_i - Z_s - V\alpha_s \\
\text{Subject to:} \quad & P_s|h_{sd}|^2 + \sum_{i\in\mathcal{U}_k}P_i|h_{id}|^2 \geq \theta \\
& P_s^{\mathcal{U}_k} \leq P_s \leq P_s^{max} \\
& 0 \leq P_i \leq P_i^{max} \qquad \forall i \in \mathcal{U}_k
\end{aligned} \tag{3.16}$$

where $\theta = \frac{W}{m}(2^{Rm/W} - 1)$. The solution to the LP above has a greedy structure where we start by allocating increasing power to the nodes (including s) in decreasing order of the value of $\frac{|h_{id}|^2}{(X_i+V\beta_i)}$ (where $i \in \mathcal{U}_k \cup \{s\}$) till any constraint is met.

Therefore, for this protocol, the optimal solution to finding the cost $c_3(t)$ associated with the cooperative transmission mode in (3.7) can be computed by solving (3.16) for each $\mathcal{U}_k$ and picking the one with the least cost. It is interesting to note that if we impose a constraint on the sum total power of the relays instead of individual node constraints, then due to the greedy nature of the solution to (3.16), it is optimal to select at most 1 relay for cooperation. Specifically, this relay is the one that has the highest value of $\frac{|h_{id}|^2}{(X_i+V\beta_i)}$.

3.5.2 Non-Regenerative DF, Orthogonal Channels

This protocol is similar to Regenerative DF protocol discussed in Sec. 3.5.1. The only difference is that here, in the second stage, the subset $\mathcal{U} \subset \mathcal{R}$ relays that were successful in reliably decoding the packet re-encode it using *independent* code books. In this case, the total mutual information is given by [LW03]:

$$\frac{W}{m} \log\left(1 + \frac{mP_s}{W}|h_{sd}|^2\right) + \sum_{i \in \mathcal{R}} \frac{W}{m} \log\left(1 + x_i \frac{mP_i}{W}|h_{id}|^2\right)$$

Using the same definition of binary variables x_i as in Sec.3.5.1 , we can express (3.7) for this protocol as an optimization problem that resembles (3.14). Similar to the Regenerative DF case, we can then reduce this to a set of $m+1$ subproblems, one for each $\mathcal{U}_k$. The subproblem for set $\mathcal{U}_k$ is given by:

$$\begin{aligned}
\text{Minimize:} \quad & (X_s + V\beta_s)P_s + \sum_{i \in \mathcal{U}_k} (X_i + V\beta_i)P_i - Z_s - V\alpha_s \\
\text{Subject to:} \quad & \log\left(1 + \frac{mP_s}{W}|h_{sd}|^2\right) + \sum_{i \in \mathcal{U}_k} \log\left(1 + \frac{mP_i}{W}|h_{id}|^2\right) \geq \frac{mR}{W} \\
& P_s^{\mathcal{U}_k} \leq P_s \leq P^{max} \\
& 0 \leq P_i \leq P^{max} \qquad \forall i \in \mathcal{U}_k
\end{aligned} \tag{3.17}$$

The above problem is convex and we can use the KKT conditions to get the optimal solution (see Appendix B.2 for details). Define $[x]_0^{P^{max}} \triangleq \min[\max(x,0), P^{max}]$. Then the solution to the subproblem for set $\mathcal{U}_k$ is given by:

$$\begin{aligned} P_s^*(\mathcal{U}_k) &= \Big[\frac{\nu^*}{X_s + V\beta_s} - \frac{W}{m|h_{sd}|^2}\Big]_{P_s^{\mathcal{U}_k}}^{P_s^{max}} \\ P_i^*(\mathcal{U}_k) &= \Big[\frac{\nu^*}{X_i + V\beta_i} - \frac{W}{m|h_{id}|^2}\Big]_0^{P_i^{max}} \ \forall i \in \mathcal{U}_k \end{aligned} \tag{3.18}$$

where $\nu^* \geq 0$ is chosen so that the total mutual information constraint is met with equality. Therefore, the optimal solution for the cost $c_3(t)$ in (3.7) for this protocol can be computed by solving (3.18) for each $\mathcal{U}_k$ and picking one with the least cost. We note that the solution above has a water-filling type structure that is typical of related resource allocation problems in static settings.

3.5.3 AF, Orthogonal Channels

In this protocol, the source and relays are again assigned an orthogonal channel of equal size. An example slot structure is shown in Fig. 3.1(c). However, instead of trying to decode the packet, the relays amplify and forward the received signal from the first stage. The total mutual information under this protocol is given by [MY10] [ZAL07]:

$$\frac{W}{m}\log\left(1 + \frac{mP_s}{W}\Big(|h_{sd}|^2 + \sum_{i\in\mathcal{R}}\psi_i\Big)\right)$$

where $\psi_i \triangleq \frac{P_i|h_{si}|^2|h_{id}|^2}{P_s|h_{si}|^2+P_i|h_{id}|^2+W/m}$. Using this, we can express (3.7) for this model as follows.

$$\begin{aligned}
\text{Minimize:} \quad & (X_s + V\beta_s)P_s + \sum_{i\in\mathcal{R}}(X_i + V\beta_i)P_i - Z_s - V\alpha_s \\
\text{Subject to:} \quad & \frac{W}{m}\log\left(1 + \frac{mP_s}{W}\left(|h_{sd}|^2 + \sum_{i\in\mathcal{R}}\psi_i\right)\right) \geq R \\
& 0 \leq P_s \leq P_s^{max} \\
& 0 \leq P_i \leq P_i^{max} \; \forall i \in \mathcal{R} \qquad (3.19)
\end{aligned}$$

This problem is non-convex. However, if we fix the source power P_s, then it becomes convex in the other variables. This reduction has been used in [ZAL07] as well, although it considers a static scenario with the objective of minimizing instantaneous outage probability. After fixing P_s, we can compute the optimal relay powers for this value of P_s by solving the following:

$$\begin{aligned}
\text{Minimize:} \quad & \sum_{i\in\mathcal{R}}(X_i + V\beta_i)P_i - Z_s - V\alpha_s \\
\text{Subject to:} \quad & P_s|h_{sd}|^2 + \sum_{i\in\mathcal{R}} P_s\psi_i \geq \theta \\
& 0 \leq P_i \leq P_i^{max} \qquad \forall i \in \mathcal{R} \qquad (3.20)
\end{aligned}$$

where $\theta = \frac{W}{m}(2^{Rm/W} - 1)$. The first constraint can be simplified as:

$$P_s|h_{sd}|^2 + \sum_{i\in\mathcal{R}} P_s\psi_i = P_s(|h_{sd}|^2 + \sum_{i\in\mathcal{R}}|h_{si}|^2) - \sum_{i\in\mathcal{R}} \frac{P_s^2|h_{si}|^4 + P_s|h_{si}|^2W/m}{P_s|h_{si}|^2+P_i|h_{id}|^2+W/m}$$

Since we have fixed P_s, we can express (3.20) as:

$$\begin{aligned} \text{Minimize:} \quad & \sum_{i\in\mathcal{R}} (X_i + V\beta_i)P_i - Z_s - V\alpha_s \\ \text{Subject to:} \quad & \sum_{i\in\mathcal{R}} \frac{P_s^2|h_{si}|^4 + P_s|h_{si}|^2 W/m}{P_s|h_{si}|^2 + P_i|h_{id}|^2 + W/m} \leq \theta' \\ & 0 \leq P_i \leq P_i^{max} \qquad \forall i \in \mathcal{R} \end{aligned} \tag{3.21}$$

where $\theta' = P_s(|h_{sd}|^2 + \sum_{i\in\mathcal{R}_s} |h_{si}|^2) - \theta$. Using the KKT conditions, the solution the above convex optimization problem is given by (see Appendix B.3 for details): $P_i^* = \left[\sqrt{\frac{\nu^*(P_s^2|h_{si}|^4 + P_s|h_{si}|^2 W/m)}{(X_i+V\beta_i)|h_{id}|^2}} - \frac{P_s|h_{si}|^2+W/m}{|h_{id}|^2}\right]_0^{P_i^{max}}$ where $\nu^* \geq 0$ is chosen so that the second constraint is met with equality. We note that this solution has a water-filling type structure as well. Therefore, to compute the optimal solution to (3.7) for this protocol, we would have to solve the above for each value of $P_s \in [0, P_s^{max}]$. In practice, this computation can be simplified by considering only a discrete set of values for P_s. Because we have derived a simple closed form expression for each P_s, it is easy to compare these values over, say, a discrete list of 100 options in $[0, P_s^{max}]$ to pick the best one, which enables a very accurate approximation to optimality in real time.

3.5.4 DF with DSTC

In this protocol, all the cooperating relays in the second stage use an appropriate distributed space-time code (DSTC) [LW03] so that they can transmit simultaneously on the same channel. The slot structure under this scheme is shown in Fig.3.1(d). Suppose in the first phase of the protocol, s transmits the packet in the first half of the slot using power P_s. In the second phase, a subset $\mathcal{U} \subset \mathcal{R}$ of relays that were successful in reliably decoding the packet, re-encode it using a DSTC and transmit to the destination with power P_i (where $i \in \mathcal{U}$) in the second half of the slot. Given such a set $\mathcal{U}$, the total mutual information under this protocol is given by [LTW04]:

$$\frac{W}{2} \log \left(1 + \frac{2P_s}{W}|h_{sd}|^2 + \sum_{i \in \mathcal{U}} \frac{2P_i}{W}|h_{id}|^2\right)$$

The factor of 2 appears because only half of the slot is being used for transmission. As seen in the expression above, unlike the earlier examples, this protocol does not suffer from reduced multiplexing gains due to orthogonal channels.

We can now express (3.7) for this protocol as follows. Define binary variables x_i to be 1 if relay i can reliably decode the packet after the first stage and 0 else. Then, for this protocol, (3.7) is equivalent to the following optimization problem:

$$\begin{aligned}
\text{Minimize:} \quad & (X_s + V\beta_s)P_s + \sum_{i \in \mathcal{R}} (X_i + V\beta_i)P_i - Z_s - V\alpha_s \\
\text{Subject to:} \quad & \frac{W}{2} \log \left(1 + \frac{2P_s}{W}|h_{sd}|^2 + \sum_{i \in \mathcal{R}} x_i \frac{2P_i}{W}|h_{id}|^2\right) \geq R \\
& \frac{W}{2} \log \left(1 + \frac{2P_s}{W}|h_{si}|^2\right) \geq x_i R \\
& 0 \leq P_s \leq P_s^{max} \\
& 0 \leq P_i \leq P_i^{max}, x_i \in \{0,1\} \; \forall i \in \mathcal{R}
\end{aligned} \tag{3.22}$$

By comparing the above with (3.14), it can be seen that the computation of minimum cost under this protocol follows the same procedure as described in Sec. 3.5.1 of solving

$m+1$ subproblems, each an LP, by ordering the relays greedily and hence we do not repeat it.

3.5.5 AF with DSTC

Here, all cooperating relays use amplify and forward along with DSTC. The total mutual information under this protocol is given by:

$$\frac{W}{2}\log\left(1+\frac{2P_s}{W}\Big(|h_{sd}|^2+\sum_{i\in\mathcal{R}}\psi_i\Big)\right)$$

where $\psi_i = \frac{P_i|h_{si}|^2|h_{id}|^2}{P_s|h_{si}|^2+P_i|h_{id}|^2+W/2}$. Using this, we can express (3.7) for this model as follows.

$$\begin{aligned}
\text{Minimize:} \quad & (X_s+V\beta_s)P_s+\sum_{i\in\mathcal{R}}(X_i+V\beta_i)P_i-Z_s-V\alpha_s \\
\text{Subject to:} \quad & \frac{W}{2}\log\left(1+\frac{mP_s}{W}\Big(|h_{sd}|^2+\sum_{i\in\mathcal{R}}\psi_i\Big)\right)\geq R \\
& 0\leq P_s\leq P_s^{max} \\
& 0\leq P_i\leq P_i^{max}\ \forall i\in\mathcal{R}
\end{aligned} \tag{3.23}$$

This is similar to (3.19) and thus, we fix P_s and use a similar reduction to get a convex optimization problem whose solution can be derived using KKT conditions and is given by:

$$P_i^* = \left[\sqrt{\frac{\nu^*(P_s^2|h_{si}|^4+P_s|h_{si}|^2W/2)}{(X_i+V\beta_i)|h_{id}|^2}}-\frac{P_s|h_{si}|^2+W/2}{|h_{id}|^2}\right]_0^{P_i^{max}}$$

where $\nu^* \geq 0$ is chosen so that the constraint on the total mutual information at the destination is met with equality.

3.6 Unknown Channels, Known Statistics

We next consider the solution to (3.7) when the source does not know the current channel gains and is only aware of their statistics. In this case, (3.7) becomes a 2-stage stochastic dynamic program. For brevity, here we focus on its solution for the cooperative transmission mode.

Suppose the source uses power P_s in the first stage. Let ω denote the outcome of this transmission. This lies in a space Ω of possible network states which is assumed to be of a finite but arbitrarily large size. For example, in the DF protocol, ω might represent the set of relay nodes that received the packet successfully after the first stage as well as the mutual information accumulated so far at the destination. For AF, ω can represent the SNR value at each relay node and at the destination.

Let $J_1^*(P_s, \omega)$ be the optimal cost-to-go function for the 2-stage dynamic program (3.7) given that the source uses power P_s in the first stage and the network state is ω at the beginning of the second stage. Let J_0^* denote the optimal cost-to-go function starting from the first stage. Also, let $\mathcal{R}(\omega)$ denote the set of relay nodes that can take part in cooperative transmission when the network state in ω. We define the following probabilities. Let $f(P_s, \omega)$ be the probability that the outcome of the first stage is ω when the source uses power P_s. Also, let $g(\overrightarrow{P}_{\mathcal{R}(\omega)}, P_s, \omega)$ be the probability that the receiver gets the packet successfully when relays in $\mathcal{R}(\omega)$ use a power allocation $\overrightarrow{P}_{\mathcal{R}(\omega)}$ and the source uses power P_s. Note that these probabilities are obtained by taking expectation over all channel state realizations. We assume these are obtained from the knowledge of the channel statistics.

Using these definitions, we can now write the Bellman optimality equations [Ber07] for this dynamic program $\forall \omega \in \Omega$:

$$J_0^* = \min_{P_s} \Big[(X_s + V\beta_s)P_s + \sum_{\omega \in \Omega} f(P_s, \omega) J_1^*(P_s, \omega) \Big] \tag{3.24}$$

$$J_1^*(P_s, \omega) = \min_{\overrightarrow{P}_{\mathcal{R}(\omega)}} \Big[\sum_{i \in \mathcal{R}(\omega)} (X_i + V\beta_i)P_i - (Z_s + V\alpha_s) g(\overrightarrow{P}_{\mathcal{R}(\omega)}, P_s, \omega) \Big] \tag{3.25}$$

While this can be solved using standard dynamic programming techniques, it has a computational complexity that grows with the state space size Ω and can be prohibitive when this is large. We therefore present an alternate method based on the idea of Monte Carlo simulation.

3.6.1 Simulation Based Method

Suppose the transmitter performs the following simulation. Fix a source power P_s. Define $J_0^*(P_s)$ as the optimal cost-to-go function *given* that the source uses power P_s. Note that this is simply the expression on the right hand side of (3.24) with P_s fixed. Simulate the outcome of a transmission at this power n times independently using the values of $f(P_s, \omega)$. Let $\omega_j \in \Omega$ denote the outcome of the j^{th} simulation. For each generated outcome ω_j, compute the optimal cost-to-go function $J_1^*(P_s, \omega_j)$ by solving (3.25) (this could be done using the knowledge of $g(\overrightarrow{P}_{\mathcal{R}(\omega)}, P_s, \omega)$ either analytically or numerically).

Use this to update $J_0^{est}(P_s, n)$, which is an *estimate* of $J_0^*(P_s)$ for a given P_s after n iterations and is defined as follows:

$$J_0^{est}(P_s, n) = (X_s + V\beta_s)P_s + \frac{1}{n}\sum_{j=1}^{n} J_1^*(P_s, \omega_j) \tag{3.26}$$

We now show that, for a given P_s, $J_0^{est}(P_s, n)$ can be pushed arbitrarily close to the optimal cost-to-go function $J_0^*(P_s)$ by increasing n. Since we have fixed P_s, from (3.24), we have:

$$J_0^*(P_s) = (X_s + V\beta_s)P_s + \sum_{\omega\in\Omega} f(P_s, \omega) J_1^*(P_s, \omega)$$

Define the following indicator random variables for each simulation j and $\forall\omega \in \Omega$:

$$1_\omega(P_s, j) = \begin{cases} 1 & \text{if the outcome of simulation } j \text{ is } \omega \\ 0 & \text{else} \end{cases}$$

Note that by definition $\mathbb{E}\{1_\omega(P_s, j)\} = f(P_s, \omega)$. Therefore, we can express $J_0^{est}(P_s, n)$ in terms of these indicator variables as follows:

$$J_0^{est}(P_s, n) = (X_s + V\beta_s)P_s + \frac{1}{n}\sum_{j=1}^{n}\sum_{\omega\in\Omega} 1_\omega(P_s, j) J_1^*(P_s, \omega)$$

We note that $\left(\sum_{\omega\in\Omega} 1_\omega(P_s,j)J_1^*(P_s,\omega)\right)$ are i.i.d. random variables with mean $\mu = \sum_{\omega\in\Omega} f(P_s,\omega)J_1^*(P_s,\omega)$ and variance $\sigma^2 = \sum_{\omega\in\Omega} f(P_s,\omega)(J_1^*(P_s,\omega))^2 - \mu^2$. Using Chebyshev's inequality, we get for any $\epsilon > 0$:

$$Pr\Big[|\frac{1}{n}\sum_{j=1}^{n}\Big(\sum_{\omega\in\Omega} 1_\omega(P_s,j)J_1^*(P_s,\omega)\Big) - \mu| \geq \epsilon\Big] \leq \frac{\sigma^2}{n\epsilon^2}$$

This shows that the value of the estimate quickly converges to the optimal cost-to-go value. Thus, this method can be used to get a good estimate of the optimal cost-to-go function for a fixed value of P_s in a reasonable number of steps.

3.7 Multi-Source Extensions

In this section, we extend the basic model of Sec. 3.2 to the case when there are multiple sources in the network. Let the set of source nodes be given by $\mathcal{S}$. We consider the case when all source nodes have orthogonal channels.[3] In particular, we assume that in each slot, a medium access process $\chi(t)$ determines which source nodes get transmission opportunities. For simplicity, we assume that at most one source transmits in a slot. This models situations where there might be a pseudo-random TDMA schedule that determines a unique transmitter node every slot. It also models situations where the source nodes use a contention-resolution mechanism such as CSMA. Our model can be extended to scenarios where more than one source node can transmit, potentially over orthogonal frequency channels.

[3] For the non-orthogonal scenario, there will two sources of outages: transmission failure at the physical layer and delay violation due to contention in medium access. Hence, MAC scheduling in addition to physical layer resource allocation must be considered. This is not the focus of the current work.

Let $s(t) = s(\chi(t)) \in \mathcal{S}$ be the source node that gets a transmission opportunity in slot t. Then, the optimal resource allocation framework developed in Sec. 3.4 can be applied as follows. A virtual reliability queue is defined for each source node $s \in \mathcal{S}$ and is updated as in (3.5). Note that in slots where a source node s does not get a transmission opportunity, $\Phi_s(t) = 0$. We assume that each incoming packet gets one transmission opportunity so that the delay constraint of 1 slot per packet only measures the transmission delay and not the queueing delay that would be incurred due to contention. Similarly, a virtual power queue is maintained for each node as in (3.6) including the source nodes and relay nodes. Note that in this model, it is possible for a source node to act as a relay for another source node when it is not transmitting its own data. We denote the set of relay nodes (that includes such source nodes) in slot t as $\mathcal{R}(t)$.

Then the optimal control algorithm operates as follows. Let $\boldsymbol{Q}(t)$ denote the collection of all virtual queues in timeslot t. Every slot, given $\boldsymbol{Q}(t)$ and any channel state $\mathcal{T}(t)$, it chooses a control action $\mathcal{I}_{s(t)}$ that minimizes the following stochastic metric (for a given control parameter $V \geq 0$):

$$\begin{aligned}
\text{Minimize: } & (X_{s(t)} + V\beta_{s(t)})\mathbb{E}\left\{P_{s(t)}|\boldsymbol{Q}(t), \mathcal{T}(t)\right\} + \sum_{i \in \mathcal{R}(t)} (X_i(t) + V\beta_i)\mathbb{E}\left\{P_i(t)|\boldsymbol{Q}(t), \mathcal{T}(t)\right\} \\
& - (Z_{s(t)} + V\alpha_{s(t)})\mathbb{E}\left\{\Phi_{s(t)}|\boldsymbol{Q}(t), \mathcal{T}(t)\right\} \\
\text{Subject to: } & 0 \leq P_{s(t)} \leq P_{s(t)}^{max} \\
& 0 \leq P_i(t) \leq P_i^{max} \ \forall i \in \mathcal{R}(t) \\
& \mathcal{I}_{s(t)} \in \mathcal{C} \qquad (3.27)
\end{aligned}$$

This problem can be solved using the techniques described for the single source case.

3.8 Simulations

We simulate the dynamic control algorithm (3.7) in an ad-hoc network with 3 stationary sources and 7 mobile relays as shown in Fig. 3.2. Every slot, the sources receive new packets destined for the base station according to an i.i.d. Bernoulli process of rate λ and each packet has a delay constraint of 1 slot. The sources are assumed to have orthogonal channels and can transmit either directly or cooperatively with a subset of the relays in their vicinity. We impose a cell-partitioned structure so that a source can only cooperate with the relays that are in the same cell in that slot. The relays move from one cell to the other according to a Markovian random walk. In the simulation, at the end of every slot, a relay decides to stay in its current cell with probability 0.8, else decides to move to an adjacent cell with probability 0.2 (where any of the feasible adjacent cells are equally likely).

We assume a Rayleigh fading model. The amplitude squares of the instantaneous gains on the links involving a source, the set of relays in its cell in that slot and the base station are exponentially distributed random variables with mean 1. All power values are normalized with respect to the average noise power. All nodes have an average power constraint of 1 unit and a maximum power constraint of 10 units.

We consider the Regenerative DF cooperative protocol over orthogonal channels and implement the optimal resource allocation strategy as computed in (3.16) for this network. In the first experiment, we consider the objective of minimizing the average sum power

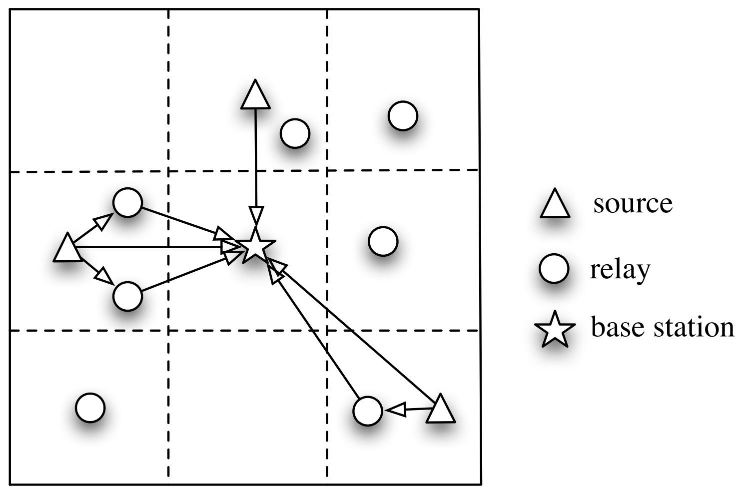

Figure 3.2: A snapshot of the example network used in simulation.

expenditure in the network given a minimum reliability constraint $\rho_s = 0.98$ and input rate $\lambda_s = 0.5$ packets/slot for all sources. For this, we set $\alpha_s = 0$ and $\beta_i = 1$. Fig. 3.3 shows the average sum power for different values of the control parameter V. It is seen that this value converges to 2.6 units for increasing values of V, as predicted by the performance bounds on the time average utility in Theorem 1. Fig. 3.4 shows the resulting average reliability queue occupancy. It is seen to increase linearly in V, again as predicted by the bound on the time average queue backlog in Theorem 1. We emphasize again that there are no actual queues in the system, and all successfully delivered packets have a delay exactly equal to 1 slot. The fact that all reliability queues are stable ensures that we are indeed meeting or exceeding the 98% reliability constraint. Indeed, in our simulations we found reliability to be almost exactly equal to the 98% constraint, as expected in an algorithm designed to minimize average power subject to this constraint. We further note that the instantaneous reliability queue value $Z(t)$ represents the worst case "excess" packets that did not meet the reliability constraints over any interval ending

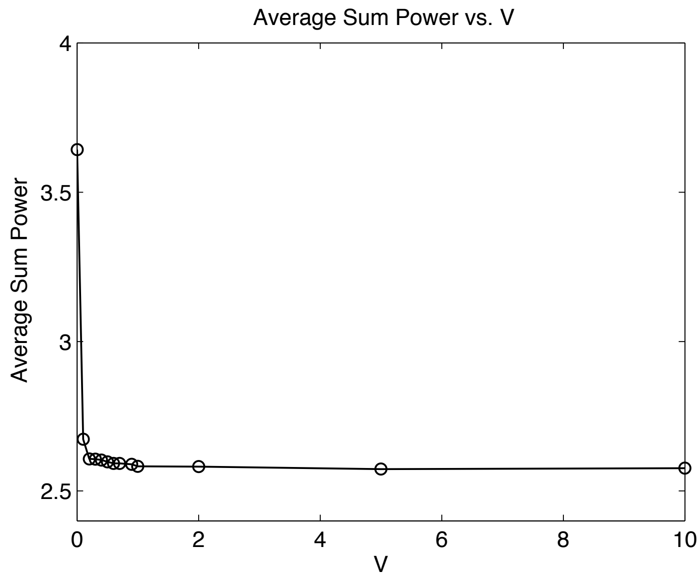

Figure 3.3: Average Sum Power vs. V.

at time t, so that maintaining small $Z(t)$ (with a small V) makes the timescales over which the time average reliability constraints are satisfied smaller.

In the second experiment, we choose both $\alpha_s = 0$ and $\beta_i = 0$ so that (3.2) becomes a feasibility problem. We fix the average and peak power values to 1 and 10 respectively and implement (3.16) for different rate-reliability pairs. In Table 3.1, we show whether these are feasible or not under three resource allocation strategies: (A) direct transmission, (B)always cooperative transmission and (C) dynamic cooperation (that corresponds to implementing the solution to (3.16) every slot). It can be seen that dynamic cooperation significantly increases the feasible rate-reliability region over direct transmission as well as static cooperation. For example, it is impossible to achieve 95% reliability using direct transmission alone, even if the traffic rate is only 0.2 packets/slot. This can be achieved by an algorithm that uses the cooperation mode (mode 3) always, but optimizes over

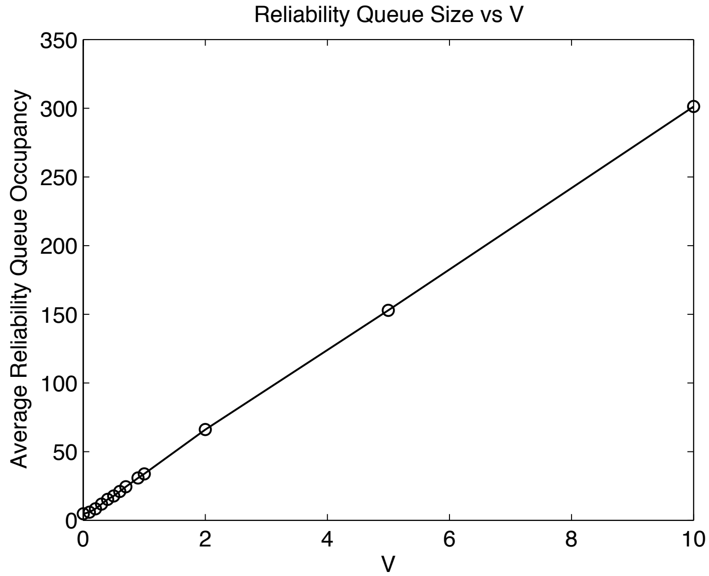

Figure 3.4: Average Reliability Queue Occupancy vs. V.

(λ_s, ρ_s)	(0.2, 0.9)	(0.2, 0.95)	(0.5, 0.95)	(0.5, 0.98)	(0.6, 0.98)	(0.7, 0.99)
A	✓	x	x	x	x	x
B	✓	✓	✓	x	x	x
C	✓	✓	✓	✓	✓	x

Table 3.1: Table showing the feasibility of different rate-reliability pairs under three strategies: (A) direct transmission, (B) always cooperate, and (C) optimal solution.

the power allocation decisions of this cooperation mode as specified in previous sections. However, always using cooperation fails if we desire 98% reliability, but using our optimal policy that dynamically mixes between the different modes, and chooses efficient power allocation decisions in each mode, can achieve 98% reliability, even at increased rates up to 0.6 packets/slot.

3.9 Chapter Summary

In this chapter, we considered the problem of optimal resource allocation for delay-limited cooperative communication in a mobile ad-hoc network. Motivated by real-time applications that have stringent delay constraints, we considered the case where each packet has a strict delay constraint of one slot. Using the technique of Lyapunov optimization, we developed dynamic cooperation strategies that make optimal use of network resources to achieve a target outage probability (reliability) for each user subject to average power constraints. Our framework is general enough to be applicable to a large class of cooperative protocols. In particular, in this chapter, we derived quasi-closed form solutions for several variants of the Decode-and-Forward and Amplify-and-Forward strategies. Unlike earlier works, our scheme does not require prior knowledge of the statistical description of the packet arrival, channel state and node mobility processes and can be implemented in an online fashion.

Chapter 4

Opportunistic Cooperation in Cognitive Networks

In this chapter, we investigate opportunistic cooperation between unlicensed secondary users and legacy primary users in a cognitive radio network. Specifically, we consider a model of a cognitive network where a secondary user can cooperatively transmit with the primary user in order to improve the latter's effective transmission rate. In return, the secondary user gets more opportunities for transmitting its own data when the primary user is idle. This kind of interaction between the primary and secondary users is different from the traditional *dynamic spectrum access* model in which the secondary users try to avoid interfering with the primary users while seeking transmission opportunities on vacant primary channels. In our model, the secondary users need to balance the desire to cooperate more (to create more transmission opportunities) with the need for maintaining sufficient energy levels for their own transmissions. Such a model is applicable in the emerging area of cognitive femtocell networks. Under these settings, we formulate the problem of maximizing the secondary user throughput subject to a time average power constraint. This is a constrained Markov Decision Problem and conventional solution techniques based on dynamic programming require either extensive knowledge of the system dynamics or learning based approaches that suffer from large convergence times. However, using the technique of Lyapunov optimization, we design a novel *greedy* and *online* control algorithm that overcomes these challenges and is provably optimal.

4.1 Introduction

Much prior work on resource allocation in cognitive radio networks has focused on the *dynamic spectrum access* model [ALVM06,ZS07,Bud07] in which the secondary users seek transmission opportunities for their packets on vacant primary channels in frequency, time, or space. Under this model, the primary users are assumed to be oblivious of the presence of the secondary users and transmit whenever they have data to send. Secondly, a collision model is assumed for the physical layer in which if a secondary user transmits on a busy primary channel, then there is a collision and both packets are lost. We considered a similar model in Chapter 2 where the objective was to design an opportunistic scheduling policy for the secondary users that maximizes their throughput utility while providing tight reliability guarantees on the maximum number of collisions suffered by a primary user over *any* given time interval. We note that this formulation does not consider the possibility of any cooperation between the primary and secondary users. Further, it is assumed that the secondary user activity does not affect the primary user channel occupancy process.

There is a growing body of work that investigates alternate models for the interaction between the primary and secondary users in a cognitive radio network. In particular, the idea of cooperation at the physical layer has been considered from an information-theoretic perspective in many works. See [GJMS09] and the references therein for a comprehensive survey. These are motivated by the work on the classical interference and relay channels [Car78, HK81, CG79, CT91]. The main idea in these works is that the resources of the secondary user can be utilized to improve the performance of the primary transmissions. In return, the secondary user can obtain more transmission opportunities on the primary channel for its own data.

These works mainly treat the problem from a physical layer/information-theoretic perspective and do not consider upper layer issues such as queueing dynamics, higher priority for primary user, etc. Recent work that addresses some of these issues includes [SBNS07, SSS+08, ZZ09, KLTM09, RE10]. Specifically, [SBNS07] considers the scenario where the secondary user acts as a relay for those packets of the primary user that it receives successfully but which are not received by the primary destination. It derives the stable throughput of the secondary user under this model. [SSS+08, ZZ09] use a Stackelberg game framework to study spectrum leasing strategies in *cooperative cognitive radio networks* where the primary users lease a portion of their licensed spectrum to secondary users in return for cooperative relaying. [KLTM09, RE10] study and compare different physical layer strategies for relaying in such cognitive cooperative systems. An important consequence of this interaction between the primary and secondary users is that the secondary user activity can now potentially influence the primary user channel occupancy process. However, there has been little work in studying this scenario. Exceptions include the work in [LMZ10] that considers a two-user setting where collisions caused by the opportunistic transmissions of the secondary user result in retransmissions by the primary user.

In this chapter, we study the problem of opportunistic cooperation in cognitive networks from a *network utility maximization* perspective, specifically taking into account the above mentioned higher-layer aspects. To motivate the problem and illustrate the design issues involved, we first consider a simple network consisting of one primary and one secondary user and their respective access points in Sec. 4.2. This can model a practical scenario of recent interest, namely a cognitive femtocell [GBA10, JL10, XL10, SR09], as discussed in Sec. 4.2. We assume that the secondary user can cooperatively transmit with the primary user to increase its transmission success probability. In return, the secondary user can get more opportunities for transmitting its own data when the primary user is idle. We formulate the problem of maximizing the secondary user throughput subject to time average power constraints in Sec. 4.2.2. Unlike most of the prior work on resource allocation in cognitive radio networks, the evolution of the system state for this problem depends on the control actions taken by the secondary user. Here, the system state refers to the channel occupancy state of the primary user. Because of this dependence, the greedy "drift-plus-penalty" minimization technique of Lyapunov optimization [GNT06] that we used in Chapters 2 and 3 is no longer optimal. Such problems are typically tackled using Markov Decision Theory and dynamic programming [Alt99, Ber07]. For example, [LMZ10] uses these tools to derive structural results on optimal channel access strategies in a similar two-user setting where collisions caused by the opportunistic transmissions of the secondary user cause the primary user to retransmit its packets. However, this approach requires either extensive knowledge of the dynamics of the underlying network state (such as state transition probabilities) or learning based approaches that suffer from large convergence times.

Instead, in Sec. 4.3, we use the recently developed framework of maximizing the *ratio* of the expected total reward over the expected length of a renewal frame [LN10, Nee10a, Nee10b] to design a control algorithm. This framework extends the classical Lyapunov optimization method [GNT06] to tackle a more general class of MDP problems where the system evolves over renewals and where the length of a renewal frame can be affected by the control decisions during that period. The resulting solution has the following structure: Rather than minimizing a "drift-plus-penalty" term every slot, it minimizes a "drift-plus-penalty ratio" over each renewal frame. This can be achieved by solving a sequence of unconstrained *stochastic shortest path* (SSP) problems and implementing the solution over every renewal frame.

While solving such SSP problems can be simpler than the original constrained MDP, it may still require knowledge of the dynamics of the underlying network state. Learning based techniques for solving such problems by sampling from the past observations have been considered in [Nee09]. However, these may suffer from large convergence times. Remarkably, in Sec. 4.4, we show that for our problem, the "drift-plus-penalty ratio" method results in an online control algorithm that *does not require any knowledge of the network dynamics or explicit learning*, yet is optimal. In this respect, it is similar to the traditional greedy "drift-plus-penalty" minimizing algorithms of Chapters 2 and 3. We then extend the basic model to incorporate multiple secondary users as well as time-varying channels in Sec. 4.6. Finally, we present simulation results in Sec. 4.7.

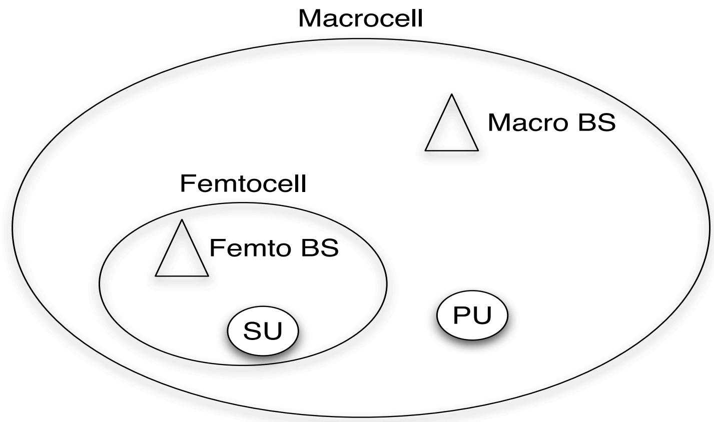

Figure 4.1: Example femtocell network with primary and secondary users.

4.2 Basic Model

We consider a network with one primary user (PU), one secondary user (SU) and their respective base stations (BS). The primary user is the licensed owner of the channel while the secondary user tries to send its own data opportunistically when the channel is not being used by the primary user. This model can capture a femtocell scenario where the primary user is a legacy mobile user that communicates with the macro base station over licensed spectrum (Fig. 4.1). The secondary user is the femtocell user that does not have any licensed spectrum of its own and tries to send data opportunistically to the femtocell base station over any vacant licensed spectrum. Similar models of cooperative cognitive radio networks have been considered in [SBNS07, SSS+08, ZZ09, KLTM09, RE10]. This can also model a single server queueing system with two classes of arrivals where one class has a strictly higher priority over the other class.

We consider a time-slotted model. We assume that the system operates over a frame-based structure. Specifically, the timeline can be divided into successive non-overlapping

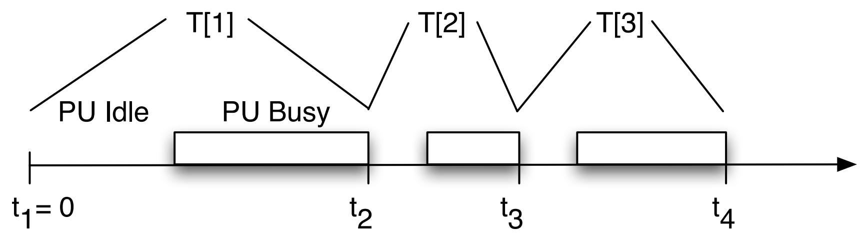

Figure 4.2: Frame-based structure of the problem under consideration. Each frame consists of two periods: PU Idle and PU Busy.

frames of duration $T[k]$ slots where $k \in \{1, 2, 3, \ldots\}$ represents the frame number (see Fig. 4.2). The start time of frame k is denoted by t_k with $t_1 = 0$. The length of frame k is given by $T[k] \triangleq t_{k+1} - t_k$. For each k, the frame length $T[k]$ is a random function of the control decisions taken during that frame. Each frame can be further divided into two periods: PU Idle and PU Busy. The "PU Idle" period corresponds to the slots when the primary user does not have any packet to send to its base station and is idle. The "PU Busy" period corresponds to the slots when the primary user is transmitting its packets to its base station over the licensed spectrum. As shown in Fig. 4.2, every frame starts with the "PU Idle" period which is followed by the "PU Busy" period and ends when the primary user becomes idle again. In the basic model, we assume that the primary user receives new packets every slot according to an i.i.d. Bernoulli arrival process $A_{pu}(t)$ with rate λ_{pu} packets/slot. This means that the length of the "PU Idle" period of any frame is a geometric random variable with parameter λ_{pu}. However, the length of the "PU Busy" period depends on the secondary user control decisions as discussed below.

In any slot t, if the primary user has a non-zero queue backlog, it transmits one packet to its base station. We assume that the transmission of each packet takes one slot. If the transmission is successful, the packet is removed from the primary user queue. However, if the transmission fails, the packet is retained in the queue for future retransmissions. The secondary user cannot transmit its packets when the channel is being used by the primary user. It can transmit its packets only during the "PU Idle" period of the frame and must stop its transmission whenever the primary user becomes active again. However, the secondary user can transmit cooperatively with the primary user in the "PU Busy" period to increase its transmission success probability. This has the effect of decreasing the expected length of the "PU Busy" period. In order to cooperate, the secondary user must allocate its power resources to help relay the primary user packet. This cooperation can take place in several ways depending on the cooperative protocol being used (see [KLTM09] for some examples). In this simple model, these details are captured by the resulting probability of successful transmission.

The reason why the secondary user may want to cooperate is because this can potentially increase the number of time slots in the future in which the primary user does not have any data to send as compared to a non-cooperative strategy. This can create more opportunities for the secondary user to transmit its own packets. However, note that the trivial strategy of cooperating whenever possible may lead to a scenario where the secondary user does not have enough power for its own data transmission. Thus, the secondary user needs to decide whether it should cooperate or not considering these two opposing factors.

The probability of a successful primary transmission depends on the control actions such as power allocation and cooperative transmission decisions by the secondary user. This is discussed in detail in the next section. In this model, we assume that the network controller cannot control the primary user actions. However, it can control the secondary user decisions on cooperation and the associated power allocation.

4.2.1 Control Decisions and Queueing Dynamics

Let $Q_{pu}(t), Q_{su}(t) \in \{0, 1, 2, \ldots\}$ represent the primary and secondary user queues respectively in slot t. New packets arrive at the secondary user according to an i.i.d. process $A_{su}(t)$ of rate λ_{su} packets/slot respectively. We assume that there exists a finite constant A_{max} such that $A_{su}(t) \leq A_{max}$ for all t. Every slot, an admission control decision determines $R_{su}(t)$, the number of new packets to admit into the secondary user queue. Further, every slot, depending on whether the primary user is busy or idle, resource allocation decisions are made as follows. When $Q_{pu}(t) > 0$, this represents the secondary user decision on cooperative transmission and the corresponding power allocation $P_{su}(t)$. When $Q_{pu}(t) = 0$, this corresponds to the secondary user decision on its own transmission and the corresponding power allocation $P_{su}(t)$.

We assume that in each slot, the secondary user can choose its power allocation $P_{su}(t)$ from a set $\mathcal{P}$ of possible options. Further, this power allocation is subject to a long-term average power constraint P_{avg} and an instantaneous peak power constraint P_{max}. For example, $\mathcal{P}$ may contain only two options $\{0, P_{max}\}$ which represents "Remain Idle" and "Cooperate/Transmit at Full Power". As another example, $\mathcal{P} = [0, P_{max}]$ such that $P_{su}(t)$ can take any value between 0 and P_{max}.

Suppose the primary user is active in slot t and the secondary user allocates power $P(t)$ for cooperative transmission. Then the random success/failure outcome of the primary transmission is given by an indicator variable $\mu_{pu}(P(t))$ and the success probability is given by $\phi(P(t)) = \mathbb{E}\{\mu_{pu}(P(t))\}$. The function $\phi(P)$ is known to the network controller and is assumed to be non-decreasing in P. However, the value of the random outcome $\mu_{pu}(P(t))$ may not be known beforehand. Note that setting $P(t) = 0$ corresponds to a non-cooperative transmission and the success probability for this case becomes $\phi(0)$ and we denote this by ϕ_{nc}. Likewise, we denote $\phi(P_{max})$ by ϕ_c. Thus, $\phi_{nc} \leq \phi(P(t)) \leq \phi_c$ for all $P(t) \in \mathcal{P}$.

We assume that λ_{pu} is such that it can be supported even when the secondary user never cooperates, i.e., $\lambda_{pu} < \phi_{nc}$. This means that the primary user queue is stable even if there is no cooperation. Further, for all k, the frame length $T[k] \geq 1$ and there exist finite constants T_{min}, T_{max} such that under all control policies, we have:

$$1 \leq T_{min} \leq \mathbb{E}\{T[k]\} \leq T_{max}$$

Specifically, T_{min} can be chosen to be the expected frame length when the secondary user always cooperates with full power while T_{max} can be chosen to be the expected frame length when the secondary user never cooperates. Using Little's Theorem, we have that: $\frac{T_{min}}{T_{min}+1/\lambda_{pu}} = \frac{\lambda_{pu}}{\phi_c}$. Similarly, we have: $\frac{T_{max}}{T_{max}+1/\lambda_{pu}} = \frac{\lambda_{pu}}{\phi_{nc}}$.

Using these, we have:

$$T_{min} \triangleq \frac{\phi_c}{(\phi_c - \lambda_{pu})\lambda_{pu}}, \qquad T_{max} \triangleq \frac{\phi_{nc}}{(\phi_{nc} - \lambda_{pu})\lambda_{pu}} \tag{4.1}$$

Finally, there exists a finite constant D such that the expectation of the second moment of a frame size, $\mathbb{E}\left\{T^2[k]\right\}$, satisfies the following for all k, regardless of the policy:

$$\mathbb{E}\left\{T^2[k]\right\} \leq D \tag{4.2}$$

This follows from the assumption that the primary user queue is stable even if there is no cooperation. In Appendix C.3, we exactly compute such a D that satisfies (4.2).

When the primary user is idle in slot t and the secondary user allocates power $P(t)$ for its own transmission, it gets a service rate given by $\mu_{su}(P(t))$. This can represent the success probability of a secondary transmission for a Bernoulli service process. This can also be used to model more general service processes. We assume that there exists a finite constant μ_{max} such that $\mu_{su}(P) \leq \mu_{max}$ for all $P \in \mathcal{P}$.

Given these control decisions, the primary and secondary user queues evolve as follows:

$$Q_{pu}(t+1) = \max[Q_{pu}(t) - \mu_{pu}(P(t)), 0] + A_{pu}(t) \tag{4.3}$$

$$Q_{su}(t+1) = \max[Q_{su}(t) - \mu_{su}(P(t)), 0] + R_{su}(t) \tag{4.4}$$

where $R_{su}(t) \leq A_{su}(t)$.

4.2.2 Control Objective

Consider any control algorithm that makes admission control decision $R_{su}(t)$ and power allocation $P_{su}(t)$ every slot subject to the constraints described in Sec. 4.2.1. Note that if the primary queue backlog $Q_{pu}(t) > 0$, then this power is used for cooperative transmission with the primary user. If $Q_{pu}(t) = 0$, then this power is used for the secondary user's own transmission. Define the following time-averages under this algorithm:

$$\overline{R}_{su} \triangleq \lim_{t\to\infty} \frac{1}{t}\sum_{\tau=0}^{t-1} \mathbb{E}\left\{R_{su}(\tau)\right\}, \overline{P}_{su} \triangleq \lim_{t\to\infty} \frac{1}{t}\sum_{\tau=0}^{t-1} \mathbb{E}\left\{P_{su}(\tau)\right\}, \overline{\mu}_{su} \triangleq \lim_{t\to\infty} \frac{1}{t}\sum_{\tau=0}^{t-1} \mathbb{E}\left\{\mu_{su}(\tau)\right\}$$

where the expectations above are with respect to the potential randomness of the control algorithm. Assuming for the time being that these limits exist, our goal is to design a joint admission control and power allocation policy that maximizes the throughput of the secondary user subject to its average and peak power constraints and the scheduling constraints imposed by the basic model. Formally, this can be stated as a stochastic optimization problem as follows:

$$\begin{aligned} \text{Maximize: } & \overline{R}_{su} \\ \text{Subject to: } & 0 \leq R_{su}(t) \leq A_{su}(t) \ \forall t \\ & P_{su}(t) \in \mathcal{P} \ \forall t \\ & \overline{R}_{su} \leq \overline{\mu}_{su} \\ & \overline{P}_{su} \leq P_{avg} \end{aligned} \tag{4.5}$$

It will be useful to define the primary queue backlog $Q_{pu}(t)$ as the "state" for this control problem. This is because the state of this queue (being zero or nonzero) affects the control options as described before. Note that the control decisions on cooperation affect the dynamics of this queue. Therefore, problem (4.5) is an instance of a constrained

Markov decision problem [Alt99]. It is well known that in order to obtain an optimal control policy, it is sufficient to consider only the class of stationary, randomized policies that take control actions only as a function of the current system state (and independent of past history). A general control policy in this class is characterized by a stationary probability distribution over the control action set for each system state. Let υ^* denote the optimal value of the objective in (4.5). Then using standard results on constrained Markov Decision problems [Alt99, Put05, BT96, Mey08], we have the following:

Lemma 2 *(Optimal Stationary, Randomized Policy): There exists a stationary, randomized policy* STAT *that takes control decisions $R_{su}^{stat}(t), P_{su}^{stat}(t)$ every slot purely as a (possibly randomized) function of the current state $Q_{pu}(t)$ while satisfying the constraints $R_{su}^{stat}(t) \leq A_{su}(t), P_{su}^{stat}(t) \in \mathcal{P}$ for all t and provides the following guarantees:*

$$\overline{R}_{su}^{stat} = \upsilon^* \tag{4.6}$$

$$\overline{R}_{su}^{stat} \leq \overline{\mu}_{su}^{stat} \tag{4.7}$$

$$\overline{P}_{su}^{stat} \leq P_{avg} \tag{4.8}$$

where $\overline{R}_{su}^{stat}, \overline{\mu}_{su}^{stat}, \overline{P}_{su}^{stat}$ denote the time-averages under this policy.

We note that the conventional techniques to solve (4.5) that are based on dynamic programming [Ber07] require either extensive knowledge of the system dynamics or learning based approaches that suffer from large convergence times. Motivated by the recently developed extension to the technique of Lyapunov optimization in [LN10,Nee10a,Nee10b], we take an different approach to this problem in the next section.

4.3 Solution Using The "Drift-plus-Penalty" Ratio Method

Recall that the start of the k^{th} frame, t_k, is defined as the first slot when the primary user becomes idle after the "PU Busy" period of the $(k-1)^{th}$ frame. Let $Q_{su}(t_k)$ denote the secondary user queue backlog at time t_k. Also let $P_{su}(t)$ be the power expenditure incurred by the secondary user in slot t. For notational convenience, in the following we will denote $\mu_{su}(P_{su}(t))$ by $\mu_{su}(t)$ noting the dependence on $P_{su}(t)$ is implicit. Then the queueing dynamics of $Q_{su}(t_k)$ satisfies the following:

$$Q_{su}(t_{k+1}) \leq \max[Q_{su}(t_k) - \sum_{t=t_k}^{t_{k+1}-1} \mu_{su}(t), 0] + \sum_{t=t_k}^{t_{k+1}-1} R_{su}(t) \tag{4.9}$$

where $R_{su}(t)$ denotes the number of new packets admitted in slot t and t_{k+1} denotes the start of the $(k+1)^{th}$ frame. The above expression has an inequality because it may be possible to serve the packets admitted in the k^{th} frame during that frame itself.

In order to meet the time average power constraint, we make use of a virtual power queue $X_{su}(t_k)$ which evolves over frames as follows:

$$X_{su}(t_{k+1}) = \max[X_{su}(t_k) - T[k]P_{avg} + \sum_{t=t_k}^{t_{k+1}-1} P_{su}(t), 0] \tag{4.10}$$

where $T[k] = t_{k+1} - t_k$ is the length of the k^{th} frame. Recall that $T[k]$ is a (random) function of the control decisions taken during the k^{th} frame.

In order to construct an optimal dynamic control policy, we use the technique of [LN10, Nee10a, Nee10b] where a ratio of "drift-plus-penalty" is maximized over every frame. Specifically, let $\boldsymbol{Q}(t_k) = (Q_{su}(t_k), X_{su}(t_k))$ denote the queueing state of the system at the start of the k^{th} frame. As a measure of the congestion in the system, we use a Lyapunov function $L(\boldsymbol{Q}(t_k)) \triangleq \frac{1}{2}[Q_{su}^2(t_k) + X_{su}^2(t_k)]$. Define the drift $\Delta(t_k)$ as the conditional expected change in $L(\boldsymbol{Q}(t_k))$ over the frame k:

$$\Delta(t_k) \triangleq \mathbb{E}\left\{L(\boldsymbol{Q}(t_{k+1})) - L(\boldsymbol{Q}(t_k))|\boldsymbol{Q}(t_k)\right\} \tag{4.11}$$

Then, using (4.9) and (4.10), we can bound $\Delta(t_k)$ as follows:

$$\begin{aligned}\Delta(t_k) \leq B - Q_{su}(t_k)\mathbb{E}&\left\{\sum_{t=t_k}^{t_{k+1}-1}\Big[\mu_{su}(t) - R_{su}(t)\Big]|\boldsymbol{Q}(t_k)\right\} \\ - X_{su}(t_k)\mathbb{E}&\left\{T[k]P_{avg} - \sum_{t=t_k}^{t_{k+1}-1} P_{su}(t)|\boldsymbol{Q}(t_k)\right\}\end{aligned} \tag{4.12}$$

where B is a finite constant that satisfies the following for all k and $\boldsymbol{Q}(t_k)$ under any control algorithm:

$$B \geq \frac{1}{2}\mathbb{E}\left\{\Big(\sum_{t=t_k}^{t_{k+1}-1}\mu_{su}(t)\Big)^2 + \Big(\sum_{t=t_k}^{t_{k+1}-1}R_{su}(t)\Big)^2 + \Big(\sum_{t=t_k}^{t_{k+1}-1}P_{su}(t) - T[k]P_{avg}\Big)^2|\boldsymbol{Q}(t_k)\right\}$$

Using the fact that $\mu_{su}(t) \leq \mu_{max}, P_{su}(t) \leq P_{max}$ for all t, and using the fact (4.2), it follows that choosing B as follows satisfies the above:

$$B = \frac{D[\mu_{max}^2 + A_{max}^2 + (P_{max} - P_{avg})^2]}{2} \tag{4.13}$$

Adding a penalty term $-V\mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1} R_{su}(t)|\boldsymbol{Q}(t_k)\right\}$ (where $V > 0$ is a control parameter that affects a utility-delay trade-off as shown in Theorem 5) to both sides and rearranging yields:

$$\begin{aligned}\Delta(t_k) - V\mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1} R_{su}(t)|\boldsymbol{Q}(t_k)\right\} &\leq B + (Q_{su}(t_k) - V)\mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1} R_{su}(t)|\boldsymbol{Q}(t_k)\right\} \\ - X_{su}(t_k)\mathbb{E}\left\{T[k]P_{avg}|\boldsymbol{Q}(t_k)\right\} &- \mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1}\Big(Q_{su}(t_k)\mu_{su}(t) - X_{su}(t_k)P_{su}(t)\Big)|\boldsymbol{Q}(t_k)\right\}\end{aligned} \tag{4.14}$$

Minimizing the ratio of an upper bound on the right hand side of the above expression and the expected frame length over all control options leads to the following *Frame-Based-Drift-Plus-Penalty-Algorithm*. In each frame $k \in \{1, 2, 3, \ldots\}$, do the following:

1. *Admission Control*: For all $t \in \{t_k, t_k + 1, \ldots, t_{k+1} - 1\}$, choose $R_{su}(t)$ as follows:

$$R_{su}(t) = \begin{cases} A_{su}(t) & \text{if } Q_{su}(t) \leq V \\ 0 & \text{else} \end{cases} \tag{4.15}$$

2. *Resource Allocation*: Choose a policy that maximizes the following ratio:

$$\frac{\mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1}\Big(Q_{su}(t_k)\mu_{su}(t) - X_{su}(t_k)P_{su}(t)\Big)|\boldsymbol{Q}(t_k)\right\}}{\mathbb{E}\left\{T[k]|\boldsymbol{Q}(t_k)\right\}} \tag{4.16}$$

Specifically, every slot t of the frame, the policy observes the queue values $Q_{su}(t_k)$ and $X_{su}(t_k)$ at the beginning of the frame and selects a secondary user power $P_{su}(t)$ subject to the constraint $P_{su}(t) \in \mathcal{P}$ and the constraint on transmitting own data vs. cooperation depending on whether slot t is in the "PU Idle" or "PU Busy" period of the frame. This is done in such a way that the above frame-based ratio of expectations is maximized. Recall that the frame size $T[k]$ is influenced by the policy through the success probabilities that are determined by secondary user power selections. Further recall that these success probabilities are different during the "PU Idle" and "PU Busy" periods of the frame. An explicit policy that maximizes this expectation is given in the next section.

3. *Queue Update*: After implementing this policy, update the queues as in (4.4) and (4.10).

From the above, it can be seen that the admission control part (4.15) is a simple threshold-based decision that does not require any knowledge of the arrival rates λ_{su} or λ_{pu}. In the next section, we present an explicit solution to the maximizing policy for the resource allocation in (4.16) and show that, remarkably, it also does not require knowledge of λ_{su} or λ_{pu} and can be computed easily. We will then analyze the performance of the *Frame-Based-Drift-Plus-Penalty-Algorithm* in Sec. 4.5.

4.4 The Maximizing Policy of (4.16)

The policy that maximizes (4.16) uses only two numbers that we call P_0^* and P_1^*, defined as follows. P_0^* is given by the solution to the following optimization problem:

$$\begin{aligned} \text{Maximize: } & Q_{su}(t_k)\mu_{su}(P_0) - X_{su}(t_k)P_0 \\ \text{Subject to: } & P_0 \in \mathcal{P} \end{aligned} \tag{4.17}$$

Let $\theta^* \triangleq Q_{su}(t_k)\mu_{su}(P_0^*) - X_{su}(t_k)P_0^*$ denote the value of the objective of (4.17) under the optimal solution. Then, P_1^* is given by the solution to the following optimization problem:

$$\begin{aligned} \text{Minimize:} \quad & \frac{\theta^* + X_{su}(t_k)P_1}{\phi(P_1)} \\ \text{Subject to:} \quad & P_1 \in \mathcal{P} \end{aligned} \tag{4.18}$$

Note that both (4.17) and (4.18) are simple optimization problems in a single variable and can be solved efficiently. Given P_0^* and P_1^*, on every slot t of frame k, the policy that maximizes (4.16) chooses power $P_{su}(t)$ as follows:

$$P_{su}(t) = \begin{cases} P_0^* & \text{if } Q_{pu}(t) = 0 \\ P_1^* & \text{if } Q_{pu}(t) > 0 \end{cases} \tag{4.19}$$

That is, the secondary user uses the constant power P_0^* for its own transmission during the "PU Idle" period of the frame, and uses constant power P_1^* for cooperative transmission during all slots of the "PU busy" period of the frame. Note that P_0^* and P_1^* can be computed easily based on the weights $Q_{su}(t_k), X_{su}(t_k)$ associated with frame k, and do not require knowledge of the arrival rates $\lambda_{su}, \lambda_{pu}$.

Our proof that the above decisions maximize (4.16) has the following parts: First, we show that the decisions that maximize the ratio of expectations in (4.16) are the same as the optimal decisions in an equivalent infinite horizon Markov decision problem (MDP). Next, we show that the solution to the infinite horizon MDP uses fixed power P_i for each queue state $Q_{pu}(t) = i$ (for $i \in \{0, 1, 2, \ldots\}$). Then, we show that P_i are the same for all $i \geq 1$. Finally, we show that the optimal powers P_0^* and P_1^* are given as above. The detailed proof is given in the next section.

4.4.1 Proof Details

Recall that the *Frame-Based-Drift-Plus-Penalty-Algorithm* chooses a policy that maximizes the following ratio over every frame $k \in \{1, 2, 3, \ldots\}$

$$\frac{\mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1}\Big(Q_{su}(t_k)\mu_{su}(t) - X_{su}(t_k)P_{su}(t)\Big)|\boldsymbol{Q}(t_k)\right\}}{\mathbb{E}\left\{T[k]|\boldsymbol{Q}(t_k)\right\}} \tag{4.20}$$

subject to the constraints described in Sec. 4.2. Here we examine how to solve (4.20) in detail. First, define the state i in any slot $t \in \{t_k, t_k + 1, \ldots, t_{k+1} - 1\}$ as the value of the primary user queue backlog $Q_{pu}(t)$ in that slot. Now let $\mathcal{R}$ denote the class of stationary, randomized policies where every policy $r \in \mathcal{R}$ chooses a power allocation $P_i(r) \in \mathcal{P}$ in each state i according to a stationary distribution. It can be shown that it is sufficient to only consider policies in $\mathcal{R}$ to maximize (4.20). Now suppose a policy $r \in \mathcal{R}$ is implemented on a *recurrent* system with fixed $Q_{su}(t_k)$ and $X_{su}(t_k)$ and with the same state dynamics as our model. Note that $\mu_{su}(t) = 0$ for all t when the state $i \geq 1$. Then, by basic renewal theory [Gal96], we have that maximizing the ratio in (4.20) is equivalent to the following optimization problem:

$$\begin{aligned}&\text{Maximize: } Q_{su}(t_k)\mathbb{E}\left\{\mu_{su}(P_0(r))\right\}\pi_0(r) - X_{su}(t_k)\sum_{i\geq 0}\mathbb{E}\left\{P_i(r)\right\}\pi_i(r)\\ &\text{Subject to: } r \in \mathcal{R}\end{aligned} \tag{4.21}$$

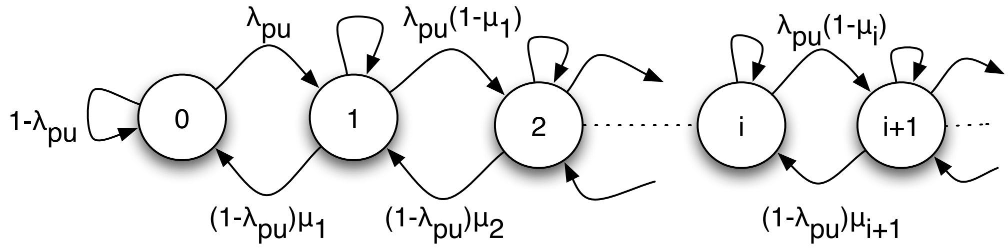

Figure 4.3: Birth-Death Markov Chain over the system state where the system state represents the primary user queue backlog.

where $\pi_i(r)$ is the resulting steady-state probability of being in state i in the recurrent system under the stationary, randomized policy r and where the expectations above are with respect to r. Note that well-defined steady-state probabilities $\pi_i(r)$ exist for all $r \in \mathcal{R}$ because we have assumed that $\lambda_{pu} < \phi_{nc}$ so that even if no cooperation is used, the primary queue is stable and the system is recurrent. Thus, solving (4.20) is equivalent to solving the *unconstrained time average maximization problem* (4.21) over the class of stationary, randomized policies. Note that (4.21) is an infinite horizon Markov decision problem (MDP) over the state space $i \in \{0, 1, 2, \ldots\}$. We study this problem in the following.

Consider the optimal stationary, randomized policy that maximizes the objective in (4.21). Let χ_i denote the probability distribution over $\mathcal{P}$ that is used by this policy to choose a power allocation P_i in state i. Let μ_i denote the resulting effective probability of successful primary transmission in state $i \geq 1$. Then we have that $\mu_i = \mathbb{E}_{\chi_i}\{\phi(P_i)\}$ where $\phi(P_i)$ denotes the probability of successful transmission in state i when the secondary user spends power P_i in cooperative transmission with the primary user. Since the system is stable and has a well-defined steady-state distribution, we can write down the detail equations for the Markov Chain that describes the state transitions of the system as follows (See Fig. 4.3):

$$\pi_0 \lambda_{pu} = \pi_1 (1 - \lambda_{pu}) \mu_1$$

$$\pi_i \lambda_{pu} (1 - \mu_i) = \pi_{i+1} (1 - \lambda_{pu}) \mu_{i+1} \qquad \forall i \geq 1$$

where π_i denotes the steady-state probability of being in state i under this policy. Summing over all i yields:

$$\lambda_{pu} = \sum_{i \geq 1} \pi_i \mu_i \tag{4.22}$$

The average power incurred in cooperative transmissions under this policy is given by:

$$\overline{P} = \sum_{i \geq 1} \pi_i \mathbb{E}_{\chi_i} \{P_i\} \tag{4.23}$$

Now consider an alternate stationary policy that uses the following fixed distribution χ' for choosing control action P' in all states $i \geq 1$:

$$\chi' \triangleq \begin{cases} \chi_1 & \text{with probability } \frac{\pi_1}{\sum_{j \geq 1} \pi_j} \\ \chi_2 & \text{with probability } \frac{\pi_2}{\sum_{j \geq 1} \pi_j} \\ \vdots & \\ \chi_i & \text{with probability } \frac{\pi_i}{\sum_{j \geq 1} \pi_j} \\ \vdots & \end{cases} \tag{4.24}$$

Let μ' denote the resulting effective probability of a successful primary transmission in any state $i \geq 1$. Note that this is same for all states by the definition (4.24). Then, we have that:

$$\mu' = \sum_{i \geq 1} \mu_i \frac{\pi_i}{\sum_{j \geq 1} \pi_j} \tag{4.25}$$

Let π'_i denote the steady-state probability of being in state i under this alternate policy. Note that the system is stable under this alternate policy as well. Thus, using the detail equations for the Markov Chain that describes the state transitions of the system under this policy yields

$$\lambda_{pu} = \sum_{k \geq 1} \pi'_k \mu' = \sum_{k \geq 1} \pi'_k \Big(\sum_{i \geq 1} \mu_i \frac{\pi_i}{\sum_{j \geq 1} \pi_j} \Big) = \sum_{k \geq 1} \pi'_k \Big(\frac{\sum_{i \geq 1} \mu_i \pi_i}{\sum_{j \geq 1} \pi_j} \Big) = \sum_{k \geq 1} \pi'_k \Big(\frac{\lambda_{pu}}{\sum_{j \geq 1} \pi_j} \Big) \tag{4.26}$$

where we used (4.22) in the last step. This implies that $\sum_{k \geq 1} \pi'_k = \sum_{j \geq 1} \pi_j$ and therefore $\pi'_0 = \pi_0$. Also, the average power incurred in cooperative transmissions under this alternate policy is given by:

$$\overline{P}' = \sum_{k \geq 1} \pi'_k \mathbb{E}_{\chi'}\{P'\} = \sum_{k \geq 1} \pi'_k \Big(\sum_{i \geq 1} \mathbb{E}_{\chi_i}\{P_i\} \frac{\pi_i}{\sum_{j \geq 1} \pi_j} \Big) = \sum_{k \geq 1} \pi'_k \Big(\frac{\overline{P}}{\sum_{j \geq 1} \pi_j} \Big) = \overline{P} \tag{4.27}$$

where we used (4.23) in the second last step and $\sum_{k \geq 1} \pi'_k = \sum_{j \geq 1} \pi_j$ in the last step.

Thus, if we choose $\chi' = \chi_0$ in state $i = 0$ and choose χ' as defined in (4.24) in all other states, it can be seen that *the alternate policy achieves the same time average value of the objective (4.21) as the optimal policy.* This implies that to maximize (4.21), it is sufficient to optimize over the class of stationary policies that use the *same* distribution for choosing P_i for all states $i \geq 1$. Denote this class by $\mathcal{R}'$. Then for all $i > 1$, we have that $\mathbb{E}\left\{P_i(r)\right\} = \mathbb{E}\left\{P_1(r)\right\}$ for all $r \in \mathcal{R}'$. Using this and the fact that $1 - \pi_0(r) = \sum_{i \geq 1} \pi_i(r)$, (4.21) can be simplified as follows:

$$\text{Max: } [Q_{su}(t_k)\mathbb{E}\{\mu_{su}(P_0(r))\} - X_{su}(t_k)\mathbb{E}\left\{P_0(r)\right\}]\pi_0(r) - X_{su}(t_k)\mathbb{E}\left\{P_1(r)\right\}(1 - \pi_0(r))$$
$$\text{Subject to: } r \in \mathcal{R}' \tag{4.28}$$

where $\pi_0(r)$ is the resulting steady-state probability of being in state 0 and where $\mathbb{E}\left\{P_1(r)\right\}$ is the average power incurred in cooperative transmission in state $i = 1$ (same for all states $i \geq 1$). Next, note that the control decisions taken by the secondary user in state $i = 0$ do not affect the length of the frame and therefore $\pi_0(r)$. Further, the expectations can be removed. Therefore the first term in the problem above can be maximized separately as follows:

$$\begin{aligned} \text{Maximize: } & Q_{su}(t_k)\mu_{su}(P_0) - X_{su}(t_k)P_0 \\ \text{Subject to: } & P_0 \in \mathcal{P} \end{aligned} \tag{4.29}$$

This is the same as (4.17). Let P_0^* denote the optimal solution to (4.29) and let $\theta^* = Q_{su}(t_k)\mu_{su}(P_0^*) - X_{su}(t_k)P_0^*$ denote the value of the objective of (4.29) under the optimal solution. Note that we must have that $\theta^* \geq 0$ because the value of the objective when the secondary user chooses $P_0 = 0$ (i.e., stays idle) is 0. Then, (4.28) can be written as:

$$\begin{aligned}
\text{Maximize: } & \theta^* \pi_0(r) - X_{su}(t_k)\mathbb{E}\left\{P_1(r)\right\}(1 - \pi_0(r)) \\
\text{Subject to: } & r \in \mathcal{R}' \qquad (4.30)
\end{aligned}$$

The effective probability of a successful primary transmission in any state $i \geq 1$ is given by $\mathbb{E}\{\phi(P_1(r))\}$. Using Little's Theorem, we have $\pi_0(r) = 1 - \frac{\lambda_{pu}}{\mathbb{E}\{\phi(P_1(r))\}}$. Using this and rearranging the objective in (4.30) and ignoring the constant terms, we have the following equivalent problem:

$$\begin{aligned}
\text{Minimize: } & \frac{\theta^* + X_{su}(t_k)\mathbb{E}\{P_1(r)\}}{\mathbb{E}\{\phi(P_1(r))\}} \\
\text{Subject to: } & r \in \mathcal{R}' \qquad (4.31)
\end{aligned}$$

It can be shown that it is sufficient to consider only *deterministic* power allocations to solve (4.31) (see, for example, [Nee10b, Section 7.3.2]). This yields the following problem:

$$\begin{aligned}
\text{Minimize: } & \frac{\theta^* + X_{su}(t_k)P_1}{\phi(P_1)} \\
\text{Subject to: } & P_1 \in \mathcal{P} \qquad (4.32)
\end{aligned}$$

This is the same as (4.18). Note that solving this problem does not require knowledge of λ_{pu} or λ_{su} and can be solved easily for general power allocation options $\mathcal{P}$. We present an example that admits a particularly simple solution to this problem.

Suppose $\mathcal{P} = \{0, P_{max}\}$ so that the secondary user can either cooperate with full power P_{max} or not cooperate (with power expenditure 0) with the primary user. Then, the optimal solution to (4.32) can be calculated by comparing the value of its objective for $P_1 \in \{0, P_{max}\}$. This yields the following simple threshold-based rule:

$$P_1^* = \begin{cases} 0 & \text{if } X_{su}(t_k) \geq \frac{\theta^*(\phi_c - \phi_{nc})}{P_{max}\phi_{nc}} \\ P_{max} & \text{else} \end{cases} \tag{4.33}$$

We also note that this threshold can be computed without any knowledge of the input rates $\lambda_{pu}, \lambda_{su}$.

To summarize, the overall solution to (4.16) is given by the pair (P_0^*, P_1^*) where P_0^* denotes the power allocation used by the secondary user for its own transmission when the primary user is idle and P_1^* denotes the power used by the secondary user for cooperative transmission. Note that these values remain fixed for the entire duration of frame k. However, these can change from one frame to another depending on the values of the queues $Q_{su}(t_k), X_{su}(t_k)$. The computation of (P_0^*, P_1^*) can be carried out using a two-step process as follows:

1. First, compute P_0^* by solving problem (4.29). Let θ^* be the value of the objective of (4.29) under the optimal solution P_0^*.

2. Then compute P_1^* by solving problem (4.32).

It is interesting to note that in order to implement this algorithm, the secondary user does not require knowledge of the current queue backlog value of the primary user. Rather, it only needs to know the values of its own queues and whether the current slot is in the "PU Idle" or "PU Busy" part of the frame. This is quite different from the conventional solution to the MDP (4.5) which is typically a different randomized policy for each value of the state (i.e., the primary queue backlog).

4.5 Performance Analysis

To analyze the performance of the *Frame-Based-Drift-Plus-Penalty-Algorithm*, we compare its Lyapunov drift with that of the optimal stationary, randomized policy $STAT$ of Lemma 2. First, note that by basic renewal theory [Gal96], the performance guarantees provided by $STAT$ hold over every frame $k \in \{1, 2, 3, \ldots\}$. Specifically, let t_k be the start of the k^{th} frame. Suppose $STAT$ is implemented over this frame. Then the following hold:

$$\mathbb{E}\left\{\sum_{t=t_k}^{\hat{t}_{k+1}-1} R_{su}^{stat}(t)\right\} = \mathbb{E}\left\{\hat{T}[k]\right\} \upsilon^* \tag{4.34}$$

$$\mathbb{E}\left\{\sum_{t=t_k}^{\hat{t}_{k+1}-1} R_{su}^{stat}(t)\right\} \leq \mathbb{E}\left\{\sum_{t=t_k}^{\hat{t}_{k+1}-1} \mu_{su}^{stat}(t)\right\} \tag{4.35}$$

$$\mathbb{E}\left\{\sum_{t=t_k}^{\hat{t}_{k+1}-1} P_{su}^{stat}(t)\right\} \leq \mathbb{E}\left\{\hat{T}[k]\right\} P_{avg} \tag{4.36}$$

where $\hat{t}_{k+1}$ and $\hat{T}[k]$ denote the start of the $(k+1)^{th}$ frame and the length of the k^{th} frame, respectively, under the policy $STAT$. Similarly, $R_{su}^{stat}(t), P_{su}^{stat}(t), \mu_{su}^{stat}(t)$ denote the resource allocation decisions under $STAT$.

Next, we define an alternate control algorithm ALT that will be useful in analyzing the performance of the *Frame-Based-Drift-Plus-Penalty-Algorithm.*

Algorithm ALT: In each frame $k \in \{1, 2, 3, \ldots\}$, do the following:

1. *Admission Control*: For all $t \in \{t_k, t_k+1, \ldots, t_{k+1}-1\}$, choose $R_{su}(t)$ as follows:

$$R_{su}(t) = \begin{cases} A_{su}(t) & \text{if } Q_{su}(t_k) \leq V \\ 0 & \text{else} \end{cases} \tag{4.37}$$

2. *Resource Allocation*: Choose a policy that maximizes the following ratio:

$$\frac{\mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1}\Big(Q_{su}(t_k)\mu_{su}(t) - X_{su}(t_k)P_{su}(t)\Big)|\boldsymbol{Q}(t_k)\right\}}{\mathbb{E}\left\{T[k]|\boldsymbol{Q}(t_k)\right\}} \tag{4.38}$$

3. *Queue Update*: After implementing this policy, update the queues as in (4.9), (4.10).

By comparing with the *Frame-Based-Drift-Plus-Penalty-Algorithm*, it can be see that this algorithm differs only in the admission control part while the resource allocation decisions are exactly the same. Specifically, under *ALT*, the queue backlog $Q_{su}(t_k)$ at the start of the k^{th} frame is used for making admission control decisions for the entire duration of that frame. However, under the *Frame-Based-Drift-Plus-Penalty-Algorithm*, the queue backlog $Q_{su}(t)$ at the start of *each* slot is used for making admission control decisions. Note that since the length of the frame depends only on the resource allocation decisions and they are the same under the two algorithms, it follows that implementing them with the same starting backlog $\boldsymbol{Q}(t_k)$ yields the same frame lengths.

The following lemma compares the value of the second term in the Lyapunov drift bound (4.14) that corresponds to the admission control decisions under these two algorithms. Its proof is given in Appendix C.1.

Lemma 3 *Let $R_{su}^{fab}(t)$ and $R_{su}^{alt}(t)$ denote the admission control decisions made by the Frame-Based-Drift-Plus-Penalty-Algorithm and the ALT algorithm respectively for all $t \in \{t_k, t_k+1, \ldots, t_{k+1}-1\}$. Then we have:*

$$\mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1}(Q_{su}(t_k)-V)R_{su}^{alt}(t)|\boldsymbol{Q}(t_k)\right\} \geq \mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1}(Q_{su}(t_k)-V)R_{su}^{fab}(t)|\boldsymbol{Q}(t_k)\right\} - C \tag{4.39}$$

where $C \triangleq \frac{D(A_{max}+\mu_{max})A_{max}}{2}$ is a constant that does not depend on V.

We are now ready to characterize the performance of the *Frame-Based-Drift-Plus-Penalty-Algorithm.*

Theorem 5 *(Performance Theorem) Suppose the Frame-Based-Drift-Plus-Penalty-Algorithm is implemented over all frames $k \in \{1, 2, 3, \ldots\}$ with initial condition $Q_{su}(0) = 0, X_{su}(0) = 0$ and with a control parameter $V > 0$. Then, we have:*

1. *The secondary user queue backlog $Q_{su}(t)$ is upper bounded for all t:*

$$Q_{su}(t) \leq Q_{max} \triangleq A_{max} + V \tag{4.40}$$

2. *The virtual power queue $X_{su}(t_k)$ is mean rate stable, i.e.,*

$$\lim_{K\to\infty} \frac{\mathbb{E}\left\{X_{su}(t_K)\right\}}{K} = 0 \tag{4.41}$$

Further, we have:

$$\limsup_{K\to\infty}\left(\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1}(P_{su}^{fab}(t)-P_{avg})\right\}\right)\leq 0 \tag{4.42}$$

$$\limsup_{K\to\infty}\frac{\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1}P_{su}^{fab}(t)\right\}}{\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}\left\{T[k]\right\}}\leq P_{avg} \tag{4.43}$$

3. *The time-average secondary user throughput (defined over frames) satisfies the following bound for all $K > 0$:*

$$\frac{\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1}R_{su}(t)\right\}}{\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}\left\{T[k]\right\}}\geq \upsilon^{*}-O(1/V) \tag{4.44}$$

where $B = \frac{D[\mu_{max}^2+A_{max}^2+(P_{max}-P_{avg})^2]}{2}$ *and* $C = \frac{D(A_{max}+\mu_{max})A_{max}}{2}$.

Theorem 5 shows that the time-average secondary user throughput can be pushed to within $O(1/V)$ of the optimal value with a trade-off in the worst case queue backlog. By Little's Theorem, this leads to an $O(1/V, V)$ utility-delay tradeoff.

Proof 5 *Part (1): We argue by induction. First, note that (4.40) holds for $t = 0$. Next, suppose $Q_{su}(t) \leq Q_{max}$ for some $t > 0$. We will show that $Q_{su}(t+1) \leq Q_{max}$. We have two cases. First, suppose $Q_{su}(t) \leq V$. Then, by (4.9), the maximum that $Q_{su}(t)$ can increase is A_{max} so that $Q_{su}(t+1) \leq A_{max} + V = Q_{max}$. Next, suppose $Q_{su}(t) > V$. Then, the admission control decision (4.15) chooses $R_{su}(t) = 0$. Thus, by (4.9), we have that $Q_{su}(t+1) \leq Q_{su}(t) \leq Q_{max}$ for this case as well. Combining these two cases proves the bound (4.40).*

Parts (2) and (3): See Appendix C.2.

4.6 Extensions to Basic Model

We consider two extensions to the basic model of Sec. 4.2.

4.6.1 Multiple Secondary Users

Consider the scenario with one primary user as before, but with $N > 1$ secondary users. The primary user channel occupancy process evolves as before where the secondary users can transmit their own data only when the primary user is idle. However, they may cooperatively transmit with the primary user to increase its transmission success probability. In general, multiple secondary users may cooperatively transmit with the primary in one timeslot. However, for simplicity, here we assume that at most one secondary user can take part in a cooperative transmission per slot. Further, we also assume that at most one secondary user can transmit its data when the primary user is idle.

Our formulation can be easily extended to this scenario. Let $\mathcal{P}_i$ denote the set of power allocation options for secondary user i. Suppose each secondary user i is subject to average and peak power constraints $P_{avg,i}$ and $P_{max,i}$ respectively. Also, let $\phi_i(P)$ denote the success probability of the primary transmission when secondary user i spends power P in cooperative transmission. Now consider the objective of maximizing the sum total throughput of the secondary users subject to each user's average and peak power constraints and the scheduling constraints of the model. In order to apply the "drift-plus-penalty" ratio method, we use the following queues:

$$Q_i(t_{k+1}) \leq \max[Q_i(t_k) - \sum_{t=t_k}^{t_{k+1}-1} \mu_i(t), 0] + \sum_{t=t_k}^{t_{k+1}-1} R_i(t) \tag{4.45}$$

$$X_i(t_{k+1}) = \max[X_i(t_k) - T[k]P_{avg,i} + \sum_{t=t_k}^{t_{k+1}-1} P_i(t), 0] \tag{4.46}$$

where $Q_i(t_k)$ is the queue backlog of secondary user i at the beginning of the k^{th} frame, $\mu_i(t)$ is the service rate of secondary user i in slot t, $R_i(t)$ and $P_i(t)$ denote the number of new packets admitted and the power expenditure incurred by the secondary user i in slot t. Finally, t_{k+1} denotes the start of the $(k+1)^{th}$ frame and $T[k] = t_{k+1} - t_k$ is the length of the k^{th} frame as before.

Let $\boldsymbol{Q}(t_k) = (Q_1(t_k), \ldots, Q_N(t_k), X_1(t_k), \ldots, X_N(t_k))$ denote the queueing state of the system at the start of the k^{th} frame. Using a Lyapunov function $L(\boldsymbol{Q}(t_k)) \triangleq \frac{1}{2}\Big[\sum_{i=1}^{N} Q_i^2(t_k) + \sum_{i=1}^{N} X_i^2(t_k)\Big]$ and following the steps in Sec. 4.3 yields the following *Multi-User Frame-Based-Drift-Plus-Penalty-Algorithm*. In each frame $k \in \{1, 2, 3, \ldots\}$, do the following:

1. *Admission Control*: For all $t \in \{t_k, t_k+1, \ldots, t_{k+1}-1\}$, for each secondary user $i \in \{1, 2, \ldots, N\}$, choose $R_i(t)$ as follows:

$$R_i(t) = \begin{cases} A_i(t) & \text{if } Q_i(t) \leq V \\ 0 & \text{else} \end{cases} \tag{4.47}$$

   where $A_i(t)$ is the number of new arrivals to secondary user i in slot t.

2. *Resource Allocation*: Choose a policy that maximizes the following ratio:

$$\frac{\sum_{i=1}^{N} \mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1} (Q_i(t_k)\mu_i(t) - X_i(t_k)P_i(t))|\boldsymbol{Q}(t_k)\right\}}{\mathbb{E}\left\{T[k]|\boldsymbol{Q}(t_k)\right\}} \tag{4.48}$$

3. *Queue Update*: After implementing this policy, update the queues as in (4.45) and (4.46).

Similar to the basic model, this algorithm can be implemented without any knowledge of the arrival rates λ_i or λ_{pu}. Further, using the techniques developed in Sec. 4.4, it can be shown that the solution to (4.48) can be computed in two steps as follows. First, we solve the following problem for each $i \in \{1, 2, \ldots, N\}$:

$$\begin{aligned} &\text{Maximize: } Q_i(t_k)\mu_i(P) - X_i(t_k)P \\ &\text{Subject to: } P \in \mathcal{P}_i \end{aligned} \tag{4.49}$$

Let P_0^* denote the optimal solution to (4.49) achieved by user i^* and let θ^* denote the optimal objective value. This means user i^* transmits on all idle slots of frame k with power P_0^*. Next, to determine the optimal cooperative transmission strategy, we solve the following problem for each $i \in \{1, 2, \ldots, N\}$:

$$\begin{aligned} &\text{Minimize: } \frac{\theta^* + X_i(t_k)P}{\phi_i(P)} \\ &\text{Subject to: } P \in \mathcal{P}_i \end{aligned} \tag{4.50}$$

Let P_1^* denote the optimal solution to (4.50) achieved by user j^*. This means user j^* cooperatively transmits on all busy slots of frame k with power P_1^*.

4.6.2 Fading Channels

Next, suppose there is an additional channel fading process $S(t)$ that takes values from a finite set $\mathcal{S}$ in an i.i.d fashion every slot. We assume that in every slot, $\text{Prob}[S(t) = s] = q_s$ for all $s \in \mathcal{S}$. The success probability with cooperative transmission now is a function of both the power allocation and the fading state in that slot. Specifically, suppose the primary user is active in slot t and the secondary user allocates power $P(t)$ for cooperative transmission. Also suppose $S(t) = s$. Then the random success/failure outcome of the primary transmission is given by an indicator variable $\mu_{pu}(P(t), s)$ and the success probability is given by $\phi_s(P(t)) = \mathbb{E}\{\mu_{pu}(P(t), s)\}$. The function $\phi_s(P)$ is known to the network controller for all $s \in \mathcal{S}$ and is assumed to be non-decreasing in P for each $s \in \mathcal{S}$. For simplicity, we assume that the secondary user transmission rate $\mu_{su}(t)$ depends only on $P(t)$.

By applying the "drift-plus-penalty" ratio method to this extended model, we get the following control algorithm. The admission control remains the same as (4.15). The resource allocation part involves maximizing the ratio in (4.16). Using the same arguments as before in Sec. 4.4, it can be shown that maximizing this ratio is equivalent to the following optimization problem:

$$\begin{aligned}
&\text{Maximize: } Q_{su}(t_k)\mathbb{E}\left\{\mu_{su}(P_0(r))\right\}\pi_0(r) - X_{su}(t_k)\mathbb{E}\left\{P_0(r)\right\}\pi_0(r) \\
&\qquad - X_{su}(t_k)\sum_{i\geq 1}\sum_{s\in\mathcal{S}}\mathbb{E}\left\{P_{i,s}(r)\right\}\pi_{i,s}(r) \\
&\text{Subject to: } r \in \mathcal{R}
\end{aligned} \tag{4.51}$$

where $\pi_{i,s}(r)$ is the resulting steady-state probability of being in state (i, s) in the recurrent system under the stationary, randomized policy r and where the expectations above are with respect to r. We study this problem in the following.

Consider the optimal stationary, randomized policy that maximizes the objective in (4.51). Let $\chi_{i,s}$ denote the probability distribution over $\mathcal{P}$ that is used by this policy to choose a control action $P_{i,s}$ in state (i, s). Let $\mu_{i,s} = \mathbb{E}_{\chi_{i,s}}\{\phi_s(P_{i,s})\}$ denote the resulting effective probability of successful primary transmission in state (i, s) where $i \geq 1$. Since the system is stable under any stationary policy, total incoming rate = total outgoing rate. Thus, we get:

$$\lambda_{pu} = \sum_{i\geq 1}\sum_{s\in\mathcal{S}}\pi_{i,s}\mu_{i,s} \tag{4.52}$$

where $\pi_{i,s}$ denotes the steady-state probability of being in state (i, s) under this policy. Note that the system is stable and has a well-defined steady-state distribution. The average power incurred in cooperative transmissions under this policy is given by:

$$\overline{P} = \sum_{i \geq 1} \sum_{s \in \mathcal{S}} \pi_{i,s} \mathbb{E}_{\chi_{i,s}} \{P_{i,s}\} \tag{4.53}$$

Now consider an alternate stationary policy that, for each $s \in \mathcal{S}$, uses the following fixed distribution χ'_s for choosing control action P'_s in all states (i, s) where $i \geq 1$:

$$\chi'_s \triangleq \begin{cases} \chi_{1,s} & \text{with probability } \frac{\pi_{1,s}}{\sum_{j \geq 1} \pi_{j,s}} \\ \chi_{2,s} & \text{with probability } \frac{\pi_{2,s}}{\sum_{j \geq 1} \pi_{j,s}} \\ \vdots & \\ \chi_{i,s} & \text{with probability } \frac{\pi_{i,s}}{\sum_{j \geq 1} \pi_{j,s}} \\ \vdots & \end{cases} \tag{4.54}$$

For each $s \in \mathcal{S}$, let μ'_s denote the resulting effective probability of a successful primary transmission in any state (i, s) where $i \geq 1$ under this policy. Note that this is same for all states (i, s) where $i \geq 1$ by the definition (4.54). Then, we have that:

$$\mu'_s = \sum_{i \geq 1} \mu_{i,s} \frac{\pi_{i,s}}{\sum_{j \geq 1} \pi_{j,s}} \tag{4.55}$$

Let $\pi'_{i,s}$ denote the steady-state probability of being in state (i, s) under this alternate policy. Since the system is stable under any stationary policy, total incoming rate = total outgoing rate. Thus, we get:

$$\lambda_{pu} = \sum_{s\in\mathcal{S}}\sum_{k\geq 1}\pi'_{k,s}\mu'_s = \sum_{s\in\mathcal{S}}\mu'_s\left(\sum_{k\geq 1}\pi'_{k,s}\right) = \sum_{s\in\mathcal{S}}\left[\sum_{i\geq 1}\mu_{i,s}\frac{\pi_{i,s}}{\sum_{j\geq 1}\pi_{j,s}}\right]\left(\sum_{k\geq 1}\pi'_{k,s}\right) \quad (4.56)$$

where we used (4.55) in the last step. Since $S(t)$ is i.i.d., for any $s_1, s_2 \in \mathcal{S}$, we have that

$$\pi_0 q_{s1} + \sum_{j\geq 1}\pi_{j,s1} = q_{s1}, \qquad \pi_0 q_{s2} + \sum_{j\geq 1}\pi_{j,s2} = q_{s2}$$

Similarly, we have:

$$\pi'_0 q_{s1} + \sum_{j\geq 1}\pi'_{j,s1} = q_{s1}, \qquad \pi'_0 q_{s2} + \sum_{j\geq 1}\pi'_{j,s2} = q_{s2}$$

Using this, for any $s_1, s_2 \in \mathcal{S}$, we have:

$$\frac{\sum_{j\geq 1}\pi_{j,s1}}{\sum_{j\geq 1}\pi'_{j,s1}} = \frac{\sum_{j\geq 1}\pi_{j,s2}}{\sum_{j\geq 1}\pi'_{j,s2}} \quad (4.57)$$

Using this in (4.56), we have for each $\hat{s} \in \mathcal{S}$:

$$\lambda_{pu} = \left[\sum_{s\in\mathcal{S}}\sum_{i\geq 1}\mu_{i,s}\pi_{i,s}\right]\frac{\sum_{k\geq 1}\pi'_{k,\hat{s}}}{\sum_{j\geq 1}\pi_{j,\hat{s}}} = \lambda_{pu}\frac{\sum_{k\geq 1}\pi'_{k,\hat{s}}}{\sum_{j\geq 1}\pi_{j,\hat{s}}} \quad (4.58)$$

where we used (4.52) in the last step. This implies that $\sum_{k\geq 1}\pi'_{k,\hat{s}} = \sum_{j\geq 1}\pi_{j,\hat{s}}$ for every $\hat{s} \in \mathcal{S}$ and therefore $\pi'_0 = \pi_0$. Also, the average power incurred in cooperative transmissions under this alternate policy is given by:

$$\begin{aligned}
\overline{P}' &= \sum_{k\geq 1}\sum_{s\in\mathcal{S}} \pi'_{k,s}\mathbb{E}_{\chi'_s}\{P'_s\} \\
&= \sum_{k\geq 1}\sum_{s\in\mathcal{S}} \pi'_{k,s}\left(\sum_{i\geq 1}\mathbb{E}_{\chi_{i,s}}\{P_{i,s}\}\frac{\pi_{i,s}}{\sum_{j\geq 1}\pi_{j,s}}\right) \\
&= \sum_{s\in\mathcal{S}}\sum_{i\geq 1}\mathbb{E}_{\chi_{i,s}}\{P_{i,s}\}\pi_{i,s} = \overline{P} \qquad (4.59)
\end{aligned}$$

where we used the fact that $\sum_{k\geq 1}\pi'_{k,s} = \sum_{j\geq 1}\pi_{j,s}$ for all s. Thus, if we choose $\chi' = \chi_0$ in state $i = 0$ and choose χ'_s as defined in (4.54) in all states (i, s) where $i \geq 1$, it can be seen that the alternate policy achieves the same time average value of the objective (4.51) as the optimal policy. This implies that to maximize (4.51), it is sufficient to optimize over the class of stationary policies that, for each $s \in \mathcal{S}$, use the *same* distribution for choosing $P_{i,s}$ for all states (i, s) where $i \geq 1$. Denote this class by $\mathcal{R}'$. Using this and the fact that $\sum_{i\geq 1}\pi_{i,s}(r) = (1 - \pi_0(r))q_s$ for all s, (4.51) can be simplified as follows:

$$\begin{aligned}
&\text{Maximize: } [Q_{su}(t_k)\mathbb{E}\{\mu_{su}(P_0(r))\} - X_{su}(t_k)\mathbb{E}\{P_0(r)\}]\pi_0(r) \\
&\qquad - X_{su}(t_k)\sum_{s\in\mathcal{S}}\mathbb{E}\{P_s(r)\}(1 - \pi_0(r))q_s \\
&\text{Subject to: } r \in \mathcal{R}' \qquad (4.60)
\end{aligned}$$

where $\pi_0(r)$ is the resulting steady-state probability of being in state 0 and where $\mathbb{E}\{P_s(r)\}$ is the average power incurred in cooperative transmission in any state (i, s) with $i \geq 1$.

Using the same arguments as before, the solution to (4.60) can be obtained in two steps as follows. We first compute the solution to (4.29) as before. Denoting its optimal value by θ^*, (4.60) can be written as:

$$\begin{aligned} &\text{Maximize: } \theta^* \pi_0(r) - X_{su}(t_k) \sum_{s \in \mathcal{S}} \mathbb{E}\left\{P_s(r)\right\}(1 - \pi_0(r)) q_s \\ &\text{Subject to: } r \in \mathcal{R}' \end{aligned} \tag{4.61}$$

Using Little's Theorem, we have $\pi_0(r) = 1 - \frac{\lambda_{pu}}{\sum_{s\in\mathcal{S}} q_s \mathbb{E}\{\phi_s(P_s(r))\}}$. Using this and rearranging the objective in (4.61) and ignoring the constant terms, we have the following equivalent problem:

$$\begin{aligned} &\text{Maximize: } \frac{-\theta^* - X_{su}(t_k) \sum_{s\in\mathcal{S}} q_s \mathbb{E}\{P_s(r)\}}{\sum_{s\in\mathcal{S}} q_s \mathbb{E}\{\phi_s(P_s(r))\}} \\ &\text{Subject to: } r \in \mathcal{R}' \end{aligned} \tag{4.62}$$

It can be shown that it is sufficient to consider only *deterministic* power allocations to solve (4.62) (see, for example, [Nee10b, Section 7.3.2]). This yields the following problem:

$$\begin{aligned} &\text{Maximize: } \frac{-\theta^* - X_{su}(t_k) \sum_{s\in\mathcal{S}} q_s P_s}{\sum_{s\in\mathcal{S}} q_s \phi_s(P_s)} \\ &\text{Subject to: } P_s \in \mathcal{P} \text{ for all } s \in \mathcal{S} \end{aligned} \tag{4.63}$$

Note that solving this problem does not require knowledge of λ_{pu} or λ_{su} and can be solved efficiently for general power allocation options $\mathcal{P}$.

4.7 Simulations

In this section, we evaluate the performance of the *Frame-Based-Drift-Plus-Penalty-Algorithm* using simulations. We consider the network model as discussed in Sec. 4.2 with one primary and one secondary user. The set $\mathcal{P}$ consists of only two options $\{0, P_{max}\}$. We assume that $P_{avg} = 0.5$ and $P_{max} = 1$. We set $\phi_{nc} = 0.6$ and $\phi_c = 0.8$. For simplicity, we assume that $\mu_{su}(P_{max}) = 1$.

In the first set of simulations, we fix the input rates $\lambda_{pu} = \lambda_{su} = 0.5$ packets/slot. For these parameters, we can compute the optimal offline solution by linear programming. This yields the maximum secondary user throughput as 0.25 packets/slot. We now simulate the *Frame-Based-Drift-Plus-Penalty-Algorithm* for different values of the control parameter V over 1000 frames. In Fig. 4.4, we plot the average throughput achieved by the secondary user over this period. It can be seen that the average throughput increases with V and converges to the optimal value 0.25 packets/slot, with the difference exhibiting a $O(1/V)$ behavior as predicted by Theorem 5. In Fig. 4.5, we plot the average queue backlog of the secondary user over this period. It can be see that the average queue backlog grows linearly in V, again as predicted by Theorem 5. Also, for all V, the average secondary user power consumption over this period was found not to exceed $P_{avg} = 0.5$ units/slot.

For comparison, we also simulate three alternate algorithms. In the first algorithm "No Cooperation", the secondary user never cooperates with the primary user and only attempts to maximize its throughput over the resulting idle periods. The secondary user throughput under this algorithm was found to be 0.166 packets/slot as shown in Fig. 4.4.

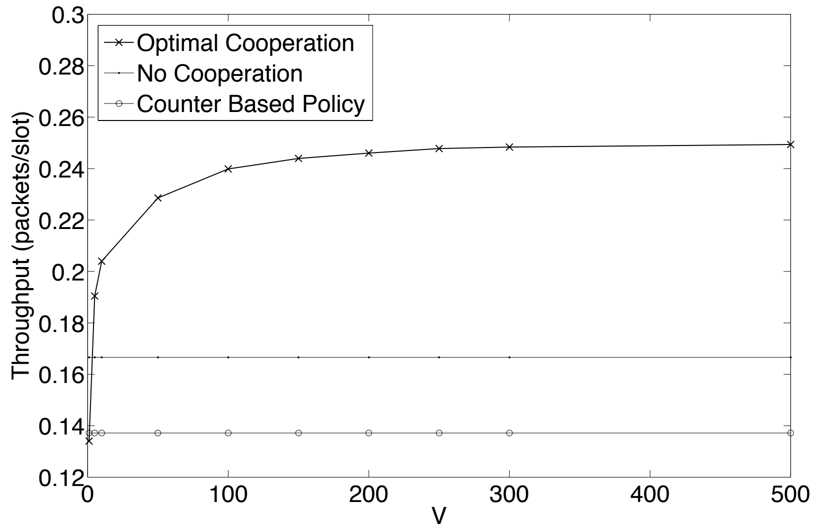

Figure 4.4: Average Secondary User Throughput vs. V.

Note that using Little's Theorem, the resulting fraction of time the primary user is idle is $1 - \lambda_{pu}/\phi_{nc} = 1 - 0.5/0.6 = 0.166$. This limits the maximum secondary user throughput under the "No Cooperation" case to 0.166 packets/slot.

In the second algorithm, we consider the "Always Cooperate" case where the secondary user always cooperates with the primary user. For the example under consideration, this uses up all the secondary user power and thus, the secondary user achieves zero throughput.

In the third algorithm "Counter Based Policy", a running average of the total secondary user power consumption so far is maintained. In each slot, the secondary user decides to transmit/cooperate only if this running average is smaller than P_{avg}. The maximum secondary user throughput under this algorithm was found to be 0.137 packets/slot. This demonstrates that simply satisfying the average power constraint is not sufficient to achieve maximum throughput. For example, it may be the case that under the "Counter

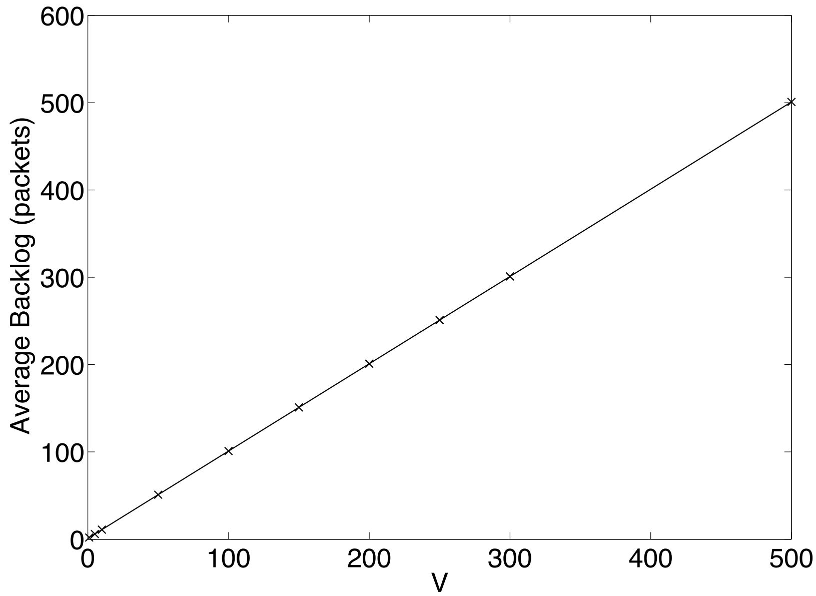

Figure 4.5: Average Secondary User Queue Occupancy vs. V.

Based Policy", the running average condition is usually satisfied when the primary user is busy. This causes the secondary user to cooperate. However, by the time the primary user next becomes idle, the running average exceeds P_{avg} so that the secondary user does not transmit its own data. In contrast, the *Frame-Based-Drift-Plus-Penalty-Algorithm* is able to find the opportune moments to cooperate/transmit optimally.

In the second set of simulations, we fix the input rate $\lambda_{su} = 0.8$ packets/slot, $V = 500$, and simulate the *Frame-Based-Drift-Plus-Penalty-Algorithm* over 1000 frames. At the start of the simulation, we set $\lambda_{pu} = 0.4$ packets/slot. The values of the other parameters remain the same. However, during the course of the simulation, we change λ_{pu} to 0.2 packets/slot after the first 350 frames and then again to 0.55 packets/slot after the first 700 frames. In Figs. 4.6 and 4.7, we plot the running average (over 100 frames) of the secondary user throughput and the average power used for cooperation. These show that the *Frame-Based-Drift-Plus-Penalty-Algorithm* automatically adapts to the changes in λ_{pu}. Further, it quickly approaches the optimal performance corresponding to the new λ_{pu} by adaptively spending more or less power (as required) on cooperation. For example, when λ_{pu} reduces to 0.2 packets/slot after frame number 350, the fraction of time the primary is idle even with no cooperation is $1-0.2/0.6 = 0.66$. With $P_{avg} = 0.5$, there is no need to cooperate anymore. This is precisely what the *Frame-Based-Drift-Plus-Penalty-Algorithm* does as shown in Fig. 4.7. Similarly, when λ_{pu} increases to 0.55 packets/slot after frame number 700, the *Frame-Based-Drift-Plus-Penalty-Algorithm* starts to spend more power on cooperative transmissions.

4.8 Chapter Summary

In this chapter, we studied the problem of opportunistic cooperation in a cognitive femtocell network. Specifically, we considered the scenario where a secondary user can cooperatively transmit with the primary user to increase its transmission success probability. In return, the secondary user can get more opportunities for transmitting its own data when the primary user is idle. A key feature of this problem is that here, the evolution of the system state depends on the control actions taken by the secondary user. This dependence makes it a constrained Markov Decision Problem traditional solutions to which require either extensive knowledge of the system dynamics or learning based approaches that suffer from large convergence times. However, using the technique of Lyapunov optimization, we designed a novel greedy and online control algorithm that overcomes these challenges and is provably optimal.

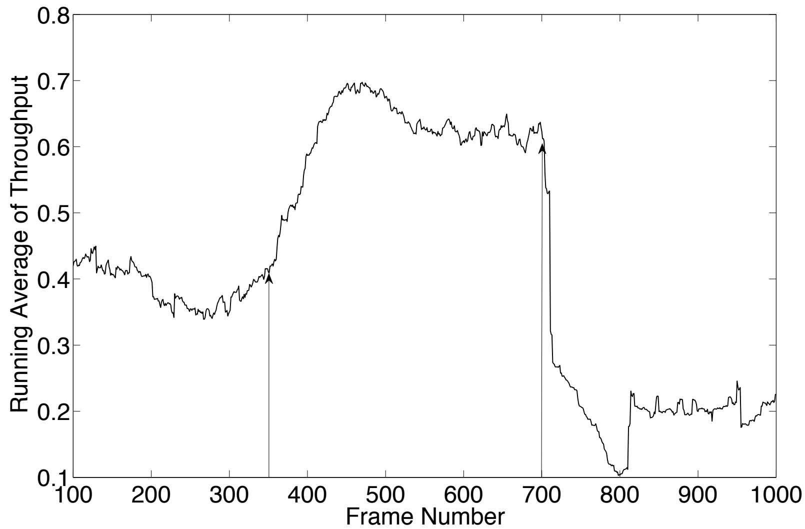

Figure 4.6: Moving Average of Secondary User Throughput over Frames.

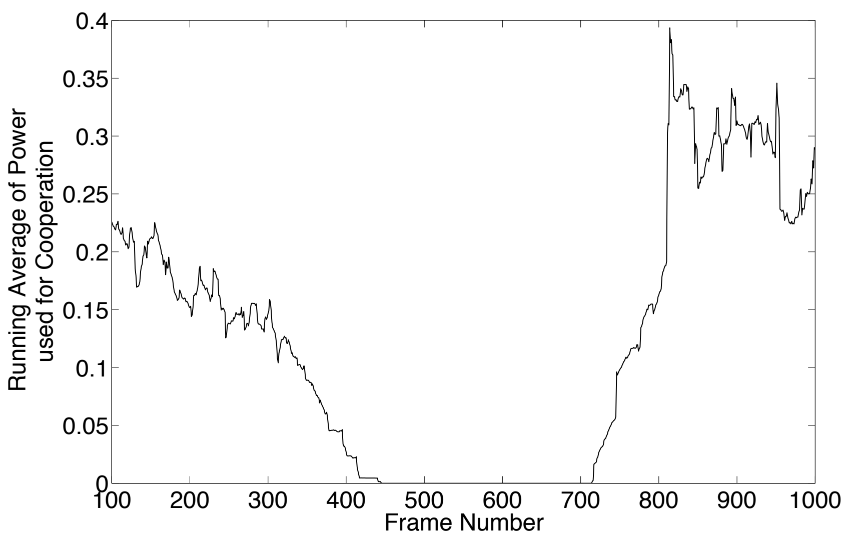

Figure 4.7: Moving Average of Power used by the Secondary User for Cooperative Transmissions over Frames.

Chapter 5

Optimal Routing with Mutual Information Accumulation

In this chapter, we investigate optimal routing and scheduling strategies for multi-hop wireless networks with *rateless codes*. Rateless codes allow each node of the network to accumulate mutual information with every packet transmission. This enables a significant performance gain over conventional shortest path routing. Further, it also outperforms cooperative communication techniques that are based on energy accumulation. However, it requires complex and combinatorial networking decisions concerning which nodes participate in transmission, and which decode ordering to use. We formulate three problems of interest in this setting: (i) minimum delay routing, (ii) minimum energy routing subject to delay constraint, and (iii) minimum delay broadcast. All of these are hard combinatorial optimization problems and we make use of several structural properties of the optimal solutions to simplify the problems and derive optimal greedy algorithms. Although the reduced problems still have exponential complexity, unlike prior works on such problems, our greedy algorithms are simple to use and do not require solving any linear programs. Further, using the insight obtained from the optimal solution to a linear network, we propose two simple heuristics that can be implemented in polynomial time in a distributed fashion and compare them with the optimal solution. Simulations suggest that both heuristics perform very close to the optimal solution over random network topologies.

5.1 Introduction

Cooperative communication promises significant gains in the performance of wireless networks over traditional techniques that treat the network as comprised of point-to-point links. Cooperative communication protocols exploit the broadcast nature of wireless transmissions and offer spatial diversity gains by making use of multiple relays for cooperative transmissions. This can increase the reliability and reduce the energy cost of data transmissions in wireless networks. See [KMY06] for a recent comprehensive survey.

Most prior work in the area of cooperative communication has investigated physical layer techniques such as orthogonal repetition coding/signaling [LTW04], distributed beamforming [MBM07], distributed space-time codes [LW03], etc. All these techniques perform *energy accumulation* from multiple transmissions to decode a packet. In energy accumulation, a receiver can decode a packet when the total received energy from multiple transmissions of that packet exceeds a certain threshold. An alternate approach of recent interest is based on *mutual information accumulation* [MMYZ07] [DLMY08]. In this approach, a node accumulates mutual information for a packet from multiple transmissions until it can be decoded successfully. This is shown to outperform energy accumulation based schemes, particularly in the high SNR regime, in [MMYZ07] [DLMY08].

Such a scheme can be implemented in practice using rateless codes of which Fountain and Raptor codes [Lub02, BLM02, Sho04] are two examples. In addition to allowing mutual information accumulation, rateless codes provide further advantages over traditional fixed rate schemes in the context of fading relay networks as discussed in [CM07] [LL09]. Unlike fixed rate code schemes in which knowledge of the current channel state information (CSI) is required at the transmitters, rateless codes adapt to the channel conditions without requiring CSI. This advantage becomes even more important in large networks where the cost of CSI acquisition grows exponentially with the network size. However, this introduces a deep memory in the system because mutual information accumulated from potentially multiple transmissions in the past can be used to decode a packet.

In this chapter, we study three problems on optimal routing and scheduling over a multi-hop wireless network using mutual information accumulation. Specifically, we first consider a network with a single source-destination pair and n relay nodes. When a node transmits, the other nodes accumulate mutual information at a rate that depends on their incoming link capacity. All nodes operate under bandwidth and energy constraints as described in detail in Sec. 2.2. We consider two problems in this setting. In the first problem, the transmit power levels of the nodes are fixed and the objective is to transmit a packet from the source to the destination in minimum delay (Sec. 5.3). In the second problem, the transmit power levels are variable and the objective is to minimize the sum total energy to deliver a packet to the destination subject to a delay constraint (Sec. 5.4). In the third problem, we consider the network model with fixed transmit power levels (similar to the first problem) and with a single source where the objective is to broadcast a packet to all the other nodes in minimum delay (Sec. 5.5). All of these objectives are important in a variety of networking scenarios.

Related problems of optimal routing in wireless networks with multi-receiver diversity have been studied in [LT06, LDFK09, NU09, DFGV10] while problems of optimal cooperative diversity routing and broadcasting are treated in [KAMZ07, SSH+10, DGG10, BK11] and references therein. Although these formulations incorporate the broadcast nature of wireless transmissions, they assume that the outcome of each transmission is a binary success/failure. Further, any packet that cannot be successfully decoded in one transmission is discarded. This is significantly different from the scenario considered in this chapter where nodes can accumulate partial information about a packet from different transmissions over time. This can be thought of as networking with "soft" information.

Prior work on accumulating partial information from multiple transmissions includes the work in [DLMY08, CJL+05, ACGW04, MY04a, SMS07, MY05, YMMZ08]. Specifically, [CJL+05] considers the problem of minimum energy unicast routing in wireless networks with energy accumulation and shows that it is an NP-complete problem. Similar results are obtained for the problem of minimum energy *accumulative broadcast* in [ACGW04, MY04a, SMS07]. A related problem of *accumulative multicast* is studied in [MY05]. Minimum energy unicast routing with energy accumulation only at the destination is considered in [YMMZ08]. Also related to the notion of accumulating partial information are the works on hydrid-ARQ techniques such as [CT01, ZV05]. The work closest to ours is [DLMY08] which treats the minimum delay routing problem with mutual information accumulation. Both [MY04a] [DLMY08] develop an LP based formulation for their respective problems that involves solving a linear program for *every* *possible ordering* of relay nodes over all subsets of relay nodes to derive the optimal solution. Thus, for a network with n relay nodes, this exhaustive approach requires solving $\sum_{m=1}^{n} \binom{n}{m} m! > n!$ linear programs.

The primary challenge associated with solving the problems addressed in this chapter is their inherent combinatorial nature. Unlike traditional shortest path routing problems, the cost of routing with mutual information accumulation depends not only on the set of nodes in the routing path, but also their *relative ordering* in the transmission sequence, making standard shortest path algorithms inapplicable. Therefore, we approach the problem differently. To derive the optimal transmission strategy for the first problem, we first formulate an optimization problem in Sec. 5.3.2 that optimizes over all possible transmission orderings over all subsets of relay nodes (similar to [MY04a] [DLMY08]). This approach clearly has a very high complexity of $O(n!)$. Then in Sec. 5.3.3, we prove a key structural property of the optimal solution that allows us to simplify the problem and derive a simple greedy algorithm that only needs to optimize over all subsets of nodes. Further, it does not require solving any linear programs. Thus, it has a complexity of $O(2^n)$. We derive a greedy algorithm of the same complexity for the second problem in Sec. 5.4. We note that this complexity, while still exponential, is a significant improvement over $O(n!)$. For example, with $n = 10$, this requires $2^{10} = 1024$ runs of a simple greedy algorithm as compared to $10! = 3628800$ runs of an LP solver. Note that for small networks, (say, $n \leq 10$), it is reasonable to use our algorithm to exactly compute the optimal solution. Further, for larger n it provides a feasible way to compute the optimal solution as a benchmark when comparing against simpler heuristics.

For the minimum delay broadcast problem, we identify a similar structural property of the optimal solution in Sec. 5.5 that allows us to simplify the problem and derive a simple greedy algorithm. While this greedy algorithm still has a complexity of $O(n!)$, it does not require solving any linear programs and thus improves over the result in [MY04a] that requires solving $n!$ linear programs. In general, we expect all these problems to be NP-complete based on the results in [CJL$^+$05, ACGW04, MY04a, SMS07].

For the special case of a line network, we derive the optimal solution in Sec. 5.3.5. Finally, in Sec. 5.6, we propose two simple heuristics that can be implemented in polynomial time in a distributed fashion and compare them with the optimal solution. Simulations suggest that both heuristics perform quite close to the optimal solution over random network topologies.

Before proceeding, we note that the techniques we apply to get these structural results can also be applied to similar problems that use energy accumulation instead of mutual information accumulation.

5.2 Network Model

The network model consists of a source s, destination d and n relays $r_1, r_2, \ldots, r_n$ as shown in Fig. 5.1. There are no time variations in the channel states. This models the scenario where the coherence time of the channels is larger than any considered transmission time of the encoded bits. In the first two problems, the source has a packet to be delivered to the destination. In the third problem, the source packet must to delivered to all nodes in the network.

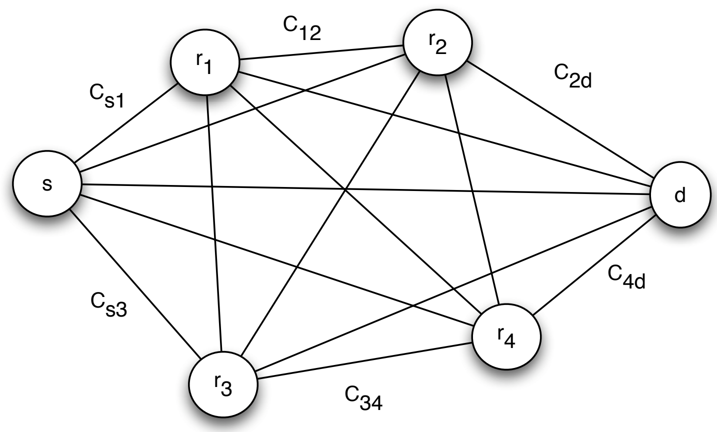

Figure 5.1: Example network with source, destination and 4 relay nodes. When a node transmits, every other node that has not yet decoded the packet accumulates mutual information at a rate given by the capacity of the link between the transmitter and that node.

Each node i transmits at a fixed power spectral density (PSD) P_i (in units of joules/sec/Hz) that is uniform across its transmission band. However, the transmission duration for a node is variable and is a design parameter. The total available bandwidth is W Hz. A node can transmit the packet only if it has fully decoded the packet. For this, it must accumulate *at least* I_{max} bits of total mutual information.

All transmissions happen on orthogonal channels in time or frequency and at most one node can transmit over a frequency channel at any given time. The channel gain between nodes i and j is given by h_{ij}. We assume a frequency non-selective, flat-fading model. Under this assumption, the minimum transmission time under the two orthogonal schemes (where nodes transmit in orthogonal time vs. frequency channels) is the same. In the following, we will focus on the case where transmissions are orthogonal in time. When a node i transmits, every other node j that does not have the full packet yet, receives mutual information at a rate that depends on the transmission capacity C_{ij} (in units of bits/sec/Hz) of link $i - j$. This transmission capacity itself depends on the transmit power and channel gain. For example, for an AWGN channel, using Shannon's formula, this is given by $C_{ij} = \log_2 \left[1 + \frac{h_{ij} P_i}{N_0}\right]$ where $N_0/2$ is the PSD of the noise process. If node i transmits for duration Δ over bandwidth W, then node j accumulates $\Delta W C_{ij}$ bits of information. In the following, we assume $W = 1$ for simplicity. We assume that each transmitting node uses independently generated *ideal* rateless codes so that the mutual information collected by a node from different transmissions add up. [1] A similar model has been considered in [DLMY08].

5.3 Minimum Delay Routing

Under the modeling assumptions discussed in Sec. 5.2, the problem of routing a packet from the source to the destination in minimum time consists of the following sub-problems:

- First, which subset of relay nodes should take part in forwarding the packet?
- Second, in what order should these nodes transmit?
- And third, what should be the transmission durations for these nodes?

We next discuss the transmission structure of a general policy under this model.

5.3.1 Timeslot and Transmission Structure

Consider any transmission strategy $\mathcal{G}$ for routing the packet to the destination in the model described above. This includes the choice of the relay set, the transmission order

[1] We can incorporate the non-idealities of the rateless codes by multiplying C_{ij} with a factor $1/(1+\epsilon)$ where $\epsilon \geq 0$ is the overhead.

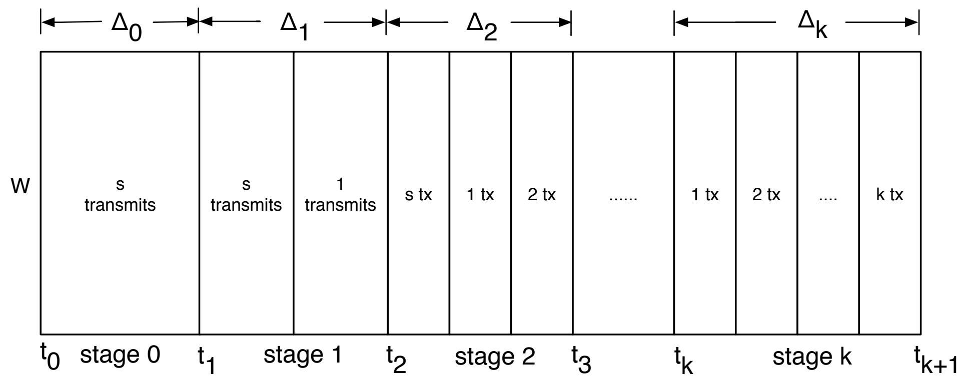

Figure 5.2: Example timeslot and transmission structure. In each stage, nodes that have already decoded the full packet transmit on orthogonal channels in time.

for this set, and the transmission durations for each node in this set. Let $\mathcal{R}$ denote the subset of relay nodes that take part in the routing process under strategy $\mathcal{G}$. By this, we mean that each node in $\mathcal{R}$ is able to decode the packet before the destination and then transmits for a non-zero duration. There could be other nodes that are able to decode the packet before the destination, but these do not take part in the forwarding process and are therefore not included in the set $\mathcal{R}$.

Let $k = |\mathcal{R}|$ be the size of this set. Also, let $\mathcal{O}$ be the ordering of nodes in $\mathcal{R}$ that describes the sequence in which nodes in $\mathcal{R}$ successfully decode the packet under strategy $\mathcal{G}$. Without loss of generality, let the relay nodes in the ordering $\mathcal{O}$ be indexed as $1, 2, 3, \ldots, k$. Also, let the source s be indexed as 0 and the destination d be indexed as $k + 1$. Initially, only the source has the packet. Let t_0 be the time when it starts its transmission and let $t_1, t_2, \ldots, t_k$ denote the times when relays $1, 2, \ldots, k$ in the ordering $\mathcal{O}$ accumulate enough mutual information to decode the packet. Also, let t_{k+1} be the time when the destination decodes the packet. By definition, $t_0 \leq t_1 \leq t_2 \leq \ldots \leq t_k \leq t_{k+1}$. We say that the transmission occurs over $k + 1$ stages, where stage $j, j \in \{0, 1, 2, \ldots, k\}$

represents the interval $[t_j, t_{j+1}]$. The state of the network at any time is given by the set of nodes that have the full packet and the mutual information accumulated so far at all the other nodes. Note that in any stage j, the first j nodes in the ordering $\mathcal{O}$ and the source have the fully decoded packet. Thus, any subset of these nodes (including potentially all of them) may transmit during this stage. Then the time-slot structure for the transmissions can be depicted as in Fig. 5.2. We note that unlike Chapter 3, here, the timeslot structure is not fixed and is part of the optimization problem. Also note that in each stage, the set of relays that have successfully decoded the packet increases by one (we ignore those relays that are not part of the set $\mathcal{R}$).

We are now ready to formulate the problem of minimum delay routing with mutual information accumulation.

5.3.2 Problem Formulation

For each j, define the duration of stage j as $\Delta_j = t_{j+1} - t_j$. Also, let A_{ij} denote the transmission duration for node i in stage j under strategy $\mathcal{G}$. Note that $A_{ij} = 0$ if $i > j$, else $A_{ij} \geq 0$. This is because node i does not have the full packet until the end of stage $i-1$. The total time to deliver the packet to the destination T_{tot} is given by $T_{tot} = t_{k+1} - t_0 = \sum_{j=0}^{k} \Delta_j$. For any transmission strategy $\mathcal{G}$ that uses the subset of relay nodes $\mathcal{R}$ with an ordering $\mathcal{O}$, the minimum delay is given by the solution to the following optimization problem:

$$
\begin{aligned}
\text{Minimize:} \quad & T_{tot} = \sum_{j=0}^{k} \Delta_j \\
\text{Subject to:} \quad & \sum_{i=0}^{m-1} \sum_{j=0}^{m-1} A_{ij} C_{im} \geq I_{max} \ \forall m \in \{1, 2, \ldots, k+1\} \\
& \sum_{i=0}^{j} A_{ij} \leq \Delta_j \ \forall j \in \{0, 1, 2, \ldots, k\} \\
& A_{ij} \geq 0 \ \forall i \in \{0, 1, 2, \ldots, k\}, j \in \{0, 1, 2, \ldots, k\} \\
& A_{ij} = 0 \ \forall i > j \\
& \Delta_j \geq 0 \ \forall j \in \{0, 1, 2, \ldots, k\}
\end{aligned} \tag{5.1}
$$

Here, the first constraint captures the requirement that node m in the ordering must accumulate at least I_{max} amount of mutual information by the end of stage $m-1$ using all transmissions in all stages up to stage $m-1$. The second constraint means that in every stage j, the total transmission time for all nodes that have the fully decoded packet in that stage cannot exceed the length of that stage. We note that the solution to (5.1) may result in a decoding order that is different from $\mathcal{O}$. In that case, the decoding order $\mathcal{O}$ is infeasible. It can be seen that the above problem is a linear program and thus can be solved efficiently for a given relay set $\mathcal{R}$ and its ordering $\mathcal{O}$. Indeed, this is the approach taken in [DLMY08] that proposes solving such a linear program for *every possible ordering of relays* for each subset of the set of relay nodes. While such an approach is guaranteed to find the optimal solution, it has a huge computational complexity of $O(n!)$ linear programs. In the next section, we show that the above computation can be significantly simplified by making use of a key structural property of the optimal solution.

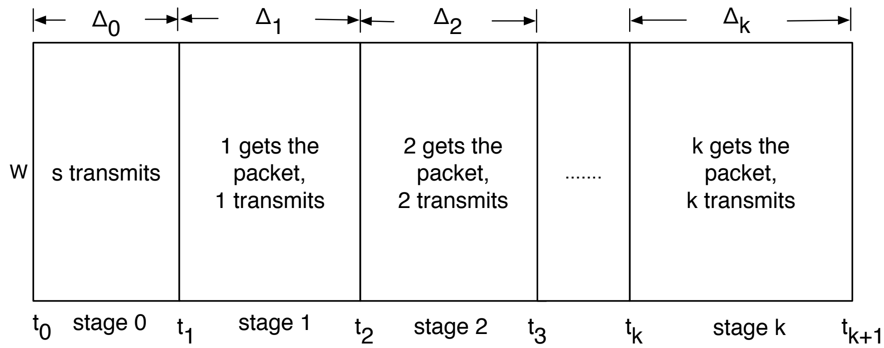

Figure 5.3: Optimal timeslot and transmission structure. In each stage, only the node that decodes the packet at the beginning of that stage transmits.

5.3.3 Characterizing the Optimal Solution of (5.1)

Let $\mathcal{R}_{opt}$ denote the subset of relay nodes that take part in the routing process in the optimal solution. Let $k = |\mathcal{R}_{opt}|$ be the size of this set. Also, let $\mathcal{O}_{opt}$ be the optimal ordering. Note that, by definition, each node in $\mathcal{R}_{opt}$ transmits for a non-zero duration (else, we can remove it from the set without affecting the minimum total transmission time). Then, we have the following:

Theorem 6 *Under the optimal solution to the minimum delay routing problem (5.1), in each stage j, it is optimal for only one node to transmit, and that node is node j.*

Fig. 5.3 shows the timeslot structure under the optimal solution. The above theorem shows that only one node transmits in each stage, and that the optimal transmission ordering is the same as the ordering that nodes in the set $\mathcal{R}_{opt}$ decode the packet. Comparing this with the general timeslot structure in Fig. 5.2, it can be seen that Theorem 6 simplifies problem (5.1) significantly. Specifically, Theorem 6 implies that, *given the optimal relay set $\mathcal{R}_{opt}$*, the optimal transmission structure (i.e., the decoding order and the transmission durations) can be computed in a *greedy* fashion as follows. First, the source starts to transmit and continues to do so until any relay node in this set gets the packet. Once this relay node gets the packet, we know from Theorem 6 that the source does not transmit in any of the remaining stages. This node then starts to transmit until another node in the set gets the packet. This process continues until the destination is able to decode the packet. The optimal solution to (5.1) can then be obtained by applying this greedy transmission strategy to all subsets of relay nodes and picking one that yields the minimum delay.[2] Note that applying this greedy transmission strategy does not require solving an LP. While searching over all subsets still has an exponential complexity of $O(2^n)$, it can be used to compute the optimal solution as a benchmark. Theorem 6 also implies that multiple copies of the packet need not be maintained across the network. For example, note that the source need not transmit after the first relay has decoded the packet and therefore can drop the packet from its queue.

We emphasize that the optimal transmission structure suggested by Theorem 6 is not obvious. For example, at the beginning of any stage, the newest addition to the set of relay nodes with the full packet may not have the best links (in terms of transmission capacity) to all the remaining nodes, including the destination. This would suggest that under the optimal solution, in general in each stage, nodes with the full packet should take turns transmitting the packet. However, Theorem 6 states that such time-sharing is not required.

Before proceeding, we present a preliminary Lemma that is used in the proof of Theorem 6. Consider any linear program:

[2]We note that the transmission structure characterized by Theorem 6 is similar to the *wavepath property* shown in [CJL$^+$05] for the problem of minimum energy unicast routing with energy accumulation in wireless networks. However, our proof technique is significantly different.

$$\begin{aligned} \text{Minimize:} \quad & c^T x \\ \text{Subject to:} \quad & Ax = b \\ & x \geq 0 \end{aligned} \tag{5.2}$$

where $x \in \mathbf{R}^n$. Then we have the following:

Lemma 4 *Let x^* be an optimal solution to the problem (5.2) such that $x^* > 0$ (where the inequality is taken entry wise). Then x^* is still an optimal solution when the constraint $x \geq 0$ is removed.*

Lemma 4 implies that removing an *inactive* constraint does not affect the optimal solution of the linear program. This is a simple fact and its proof is provided for completeness in Appendix D.1.

5.3.4 Proof of Theorem 6

Note that Theorem 6 trivially holds in stage 0 (since only the source has the full packet in this stage). Next, it is easy to see that in the last stage (i.e., stage k), only the node with the *best* link (in terms of transmission capacity) to the destination in the set $\mathcal{R}_{opt}$ should transmit in order to minimize the total delay. This is because this node will take the smallest time to transmit the remaining amount of mutual information needed by d to decode the packet. Further, we claim that this node must be the node k in the ordering $\mathcal{O}_{opt}$. This can be argued as follows. Assume that the node with the best link to the destination in the set $\mathcal{R}_{opt}$ has the full packet at some stage $(k - j)$ (where $0 < j < k$)

before the start of stage k. Then a smaller delay can be achieved by having only this node transmit after it has decoded the full packet from that stage onwards. Thus, the other nodes labeled $k-j, \ldots, k-1$ in the transmission order do not transmit, a contradiction. This shows that under the optimal solution, in the last stage k, only node k in the ordering $\mathcal{O}_{opt}$ transmits. Using induction, we now show that in every prior stage $(k-j)$ where $1 \leq j \leq k-1$, only one node needs to transmit and that this node must be node $k-j$ in the ordering $\mathcal{O}_{opt}$.

Consider the $(k-1)^{th}$ stage. At time t_{k-1}, all nodes except k and d have decoded the packet. Let the mutual information state at nodes k and d at time t_{k-1} be $I_k(t_{k-1})$ and $I_d(t_{k-1})$ respectively. Also, suppose in the $(k-1)^{th}$ stage, relay nodes $1, 2, \ldots, k-1$ and the source transmit a fraction $\alpha_1^{k-1}, \alpha_2^{k-1}, \ldots, \alpha_{k-1}^{k-1}$ and α_0^{k-1} of the total duration of stage $(k-1)$, i.e., Δ_{k-1} respectively. Note that these fractions must add to 1 since it is suboptimal to have any idle time (where no one is transmitting). Then, the optimal solution must solve the following optimization problem:

$$
\begin{aligned}
\text{Minimize:} \quad & \Delta_{k-1} + \Delta_k \\
\text{Subject to:} \quad & I_k(t_{k-1}) + \Delta_{k-1} \sum_{i=0}^{k-1} \alpha_i^{k-1} C_{ik} \geq I_{max} \\
& I_d(t_{k-1}) + \Delta_{k-1} \sum_{i=0}^{k-1} \alpha_i^{k-1} C_{id} + \Delta_k C_{kd} \geq I_{max} \\
& 0 \leq \alpha_0^{k-1}, \alpha_1^{k-1}, \ldots, \alpha_{k-1}^{k-1} \leq 1 \\
& \sum_{i=0}^{k-1} \alpha_i^{k-1} = 1 \\
& \Delta_{k-1} \geq 0, \Delta_k \geq 0
\end{aligned} \tag{5.3}
$$

Here, the first constraint states that relay k must accumulate at least I_{max} bits of mutual information by the end of stage $(k-1)$. The second constraint states that the destination must accumulate at least I_{max} bits of mutual information by the end of stage k. Note that in the last term of the left hand side of the second constraint, we have used the fact that only node k transmits during stage k.

It is easy to see that under the optimal solution, the first and second constraints must be met with equality. This simply follows from the definition of the beginning of any stage j as the time when node j has just decoded the packet. Next, let $\beta_i = \Delta_{k-1}\alpha_i^{k-1}$ for all $i \in \{0, 1, 2, \ldots, k-1\}$. Since $\sum_{i=0}^{k-1} \alpha_i^{k-1} = 1$, we have that $\sum_{i=0}^{k-1} \beta_i = \Delta_{k-1}$ and (5.3) is equivalent to:

$$
\begin{aligned}
\text{Minimize:} \quad & \sum_{i=0}^{k-1} \beta_i + \Delta_k \\
\text{Subject to:} \quad & I_k(t_{k-1}) + \sum_{i=0}^{k-1} \beta_i C_{ik} = I_{max} \\
& I_d(t_{k-1}) + \sum_{i=0}^{k-1} \beta_i C_{id} + \Delta_k C_{kd} = I_{max} \\
& \Delta_k \geq 0, \beta_i \geq 0 \qquad \forall i \in \{0, 1, 2, \ldots, k-1\}
\end{aligned}
\tag{5.4}
$$

Note that problems (5.3) and (5.4) are equivalent because we can transform (5.4) to the original problem by using the relations $\Delta_{k-1} = \sum_{i=0}^{k-1} \beta_i$ and $\alpha_i^{k-1} = \frac{\beta_i}{\Delta_{k-1}}$. The degenerate case where $\Delta_{k-1} = 0$ does not arise because if $\Delta_{k-1} = 0$, then no node transmits in stage $(k-1)$ and we transition to stage k in which only node k transmits.

This means node $k-1$ never transmits, contradicting the fact that it is part of the optimal transmission schedule.

Since we know that under the optimal solution, $\Delta_k > 0$, we can remove the constraint $\Delta_k \geq 0$ from (5.4) without affecting the optimal solution (using Lemma 4). Next we multiply the minimization objective in (5.4) by C_{kd} without changing the problem. Then, using the second equality constraint to eliminate Δ_k from the objective and ignoring the constant terms, (5.4) can be expressed as:

$$
\begin{aligned}
\text{Minimize:} \quad & \sum_{i=0}^{k-1} \beta_i (C_{kd} - C_{id}) \\
\text{Subject to:} \quad & I_k(t_{k-1}) + \sum_{i=0}^{k-1} \beta_i C_{ik} = I_{max} \\
& \beta_i \geq 0 \qquad \forall i \in \{0, 1, 2, \ldots, k-1\}
\end{aligned}
\tag{5.5}
$$

This optimization problem is linear in β_i with a single linear equality constraint and thus the solution is of the form where all except one β_i are zero. Since $\alpha_i^{k-1} = \frac{\beta_i}{\Delta_{k-1}}$, we have that in the optimal solution, exactly one of the fractions $\alpha_0^{k-1}, \alpha_1^{k-1}, \ldots, \alpha_{k-1}^{k-1}$ is equal to 1 and rest must be 0. This implies that only one node transmits in this stage. Further, this node must be the relay node $k-1$ that decoded the packet at the beginning of this stage. Else, node $k-1$ never transmits. This is because by definition of stage $(k-1)$, node $k-1$ does not have the packet before the beginning of stage $(k-1)$ and hence cannot transmit before stage $(k-1)$. Since only node k transmits when stage $(k-1)$ ends, if node $k-1$ is not the node chosen for stage $(k-1)$, it never transmits, contradicting the fact that it is part of the optimal set.[3]

Now consider the $(k-j)^{th}$ stage and suppose Theorem 6 holds for all stages after stage $(k-j)$ where $2 \leq j \leq k-1$. This means that in every stage after stage $(k-j)$, only the node that has just decoded the packet transmits. At time t_{k-j}, all nodes except $k-j+1, k-j+2, \ldots, k$ and d have decoded the packet. Let the mutual information state at these nodes at time t_{k-j} be $I_{k-j+1}(t_{k-j}), I_{k-j+2}(t_{k-j}), \ldots, I_k(t_{k-j})$ and $I_d(t_{k-j})$, respectively. Also, suppose in the $(k-j)^{th}$ stage, the source and the relay nodes $1, 2, \ldots, k-j$ transmit a fraction $\alpha_0^{k-j}, \alpha_1^{k-j}, \alpha_2^{k-j}, \ldots, \alpha_{k-j}^{k-j}$ of the total duration of stage $(k-j)$, i.e., Δ_{k-j} respectively. Then, the optimal solution must solve the following optimization problem:

$$\begin{aligned}
&\text{Minimize:} \quad && \sum_{m=0}^{j} \Delta_{k-j+m} \\
&\text{Subject to:} \quad && I_{k-j+1}(t_{k-j}) + \Delta_{k-j}\Big[\sum_{i=0}^{k-j} \alpha_i^{k-j} C_{i,k-j+1}\Big] = I_{max} \\
& && I_{k-j+n}(t_{k-j}) + \Delta_{k-j}\Big[\sum_{i=0}^{k-j} \alpha_i^{k-j} C_{i,k-j+n}\Big] + \\
& && \sum_{i=1}^{n-1} \Delta_{k-j+i} C_{k-j+i,k-j+n} = I_{max} \ \forall n \in \{2, \ldots, j+1\} \\
& && 0 \leq \alpha_0^{k-j}, \alpha_1^{k-j}, \ldots, \alpha_{k-j}^{k-j} \leq 1 \\
& && \sum_{i=0}^{k-j} \alpha_i^{k-j} = 1 \\
& && \Delta_{k-j} \geq 0, \Delta_{k-j+1} \geq 0, \ldots, \Delta_k \geq 0
\end{aligned} \tag{5.6}$$

[3] This is a crucial property that holds only for the unicast routing case. As we will see in Sec. 5.5, this does not necessarily hold for the minimum delay broadcast problem.

where the first constraint states that relay $k-j+1$ must accumulate I_{max} bits of mutual information by the end of stage $(k-j)$. The second set of constraints state that every subsequent node $k-j+n$ (where $2 \leq n \leq j+1$) including the destination in the ordering $\mathcal{O}_{opt}$ must accumulate I_{max} bits of mutual information by the end of stage $(k-j+n)$. In the last term of the left hand side of each such constraint, we have used the induction hypothesis that in every stage after stage $(k-j)$, only the node that just decoded the packet transmits. Using the transform $\beta_i = \Delta_{k-j}\alpha_i^{k-j}$ for all $i \in \{0, 1, 2, \ldots, k-j\}$, and $\sum_{i=0}^{k-j} \alpha_i^{k-j} = 1$, we have the equivalent problem:

$$
\begin{aligned}
\text{Minimize:} \quad & \sum_{i=0}^{k-j} \beta_i + \Delta_{k-j+1} + \ldots + \Delta_{k-1} + \Delta_k \\
\text{Subject to:} \quad & I_{k-j+1}(t_{k-j}) + \sum_{i=0}^{k-j} \beta_i C_{i,k-j+1} = I_{max} \\
& I_{k-j+n}(t_{k-j}) + \sum_{i=0}^{k-j} \beta_i C_{i,k-j+n} + \\
& \sum_{i=1}^{n-1} \Delta_{k-j+i} C_{k-j+i,k-j+n} = I_{max} \ \forall n \in \{2, \ldots, j+1\} \\
& \beta_i \geq 0 \ \forall i \in \{0, 1, 2, \ldots, k-j\} \\
& \Delta_{k-j+1} \geq 0, \ldots, \Delta_k \geq 0
\end{aligned}
\tag{5.7}
$$

The problems (5.6) and (5.7) are equivalent because we can transform (5.7) to the original problem by using the relations $\Delta_{k-j} = \sum_{i=0}^{k-j} \beta_i$ and $\alpha_i^{k-j} = \frac{\beta_i}{\Delta_{k-j}}$. The degenerate case where $\Delta_{k-j} = 0$ does not arise because if $\Delta_{k-j} = 0$, then no node transmits in stage $(k-j)$. We know from the induction hypothesis that only the nodes after node

$k-j$ in the ordering $\mathcal{O}_{opt}$ transmit after stage $(k-j)$. This means that node $k-j$ never transmits, a contradiction.

The second set of constraints in problem (5.7) can be written in matrix form as $\mathbf{B} + \mathbf{C}\boldsymbol{\Delta} = \mathbf{I}$ as shown below.

$$\begin{bmatrix} \sum_{i=0}^{k-j} \beta_i C_{i,k-j+2} \\ \sum_{i=0}^{k-j} \beta_i C_{i,k-j+3} \\ \vdots \\ \sum_{i=0}^{k-j} \beta_i C_{i,d} \end{bmatrix} + \begin{bmatrix} C_{k-j+1,k-j+2} & \dots & 0 \\ C_{k-j+1,k-j+3} & \dots & 0 \\ \vdots & \ddots & \vdots \\ C_{k-j+1,d} & \dots & C_{k,d} \end{bmatrix} \begin{bmatrix} \Delta_{k-j+1} \\ \Delta_{k-j+2} \\ \vdots \\ \Delta_k \end{bmatrix} = \begin{bmatrix} I_{max} - I_{k-j+2}(t_{k-j}) \\ I_{max} - I_{k-j+3}(t_{k-j}) \\ \vdots \\ I_{max} - I_d(t_{k-j}) \end{bmatrix}$$

From this, we note that $\mathbf{C}$ is a lower triangular matrix. Thus, we have: $\boldsymbol{\Delta} = \mathbf{C}^{-1}(\mathbf{I} - \mathbf{B})$. Therefore each of the terms $\Delta_{k-j+1}, \Delta_{k-j+2}, \dots, \Delta_{k-1}, \Delta_k$ is linear in the variables $\beta_0, \beta_1, \dots, \beta_{k-j}$. Using this, the objective in (5.7) can be expressed as a linear function of these variables. Let this be denoted by $f(\beta_0, \beta_1, \dots, \beta_{k-j})$. Also we know that under the optimal solution, $\Delta_{k-j+1} > 0, \dots, \Delta_k > 0$. Thus, we can remove the last set of constraints from (5.7) without affecting the optimal solution (using Lemma 4). Thus, (5.7) becomes:

$$\begin{aligned} \text{Minimize:} \quad & f(\beta_0, \beta_1, \dots, \beta_{k-j}) \\ \text{Subject to:} \quad & I_{k-j+1}(t_{k-j}) + \sum_{i=0}^{k-j} \beta_i C_{i,k-j+1} = I_{max} \\ & \beta_i \geq 0 \qquad \forall i \in \{0, 1, 2, \dots, k-j\} \end{aligned} \tag{5.8}$$

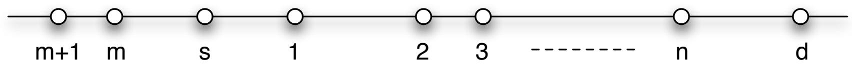

Figure 5.4: A line network.

Similar to the stage $(k-1)$ case, this optimization problem is linear in β_i with a single linear equality constraint and thus the solution is of the form where all except one β_i are zero. Since $\alpha_i^{k-j} = \frac{\beta_i}{\Delta_{k-j}}$, we have that in the optimal solution, exactly one of the fractions $\alpha_0^{k-j}, \alpha_1^{k-j}, \ldots, \alpha_{k-j}^{k-j}$ is equal to 1 and rest must be 0. This implies that only one node transmits in this stage. Further, this node must be the relay node $k-j$ that decoded the packet at the beginning of this stage. Else, node $k-j$ never transmits. This is because by definition of stage $(k-j)$, node $k-j$ does not have the packet before the beginning of stage $(k-j)$ and hence cannot transmit before stage $(k-j)$. By induction hypothesis, only nodes $k-j+1, k-j+2, \ldots, k$ transmit when stage $(k-j)$ ends. Thus, if node $k-j$ is not the node chosen for stage $(k-j)$, it never transmits, contradicting the fact that it is part of the optimal set. This proves the Theorem.

5.3.5 Exact Solution for a Line Network

In this section, we present the optimal solution for a special case of line networks. Specifically, all nodes are located on a line as shown in Fig. 5.4. We assume that each node transmits at the same PSD P. Further, the transmission capacity C_{ij} between any two nodes i and j depends only on the distance d_{ij} between the two nodes and is a monotonically decreasing function of d_{ij}. For example, we may have that $C_{ij} = \log(1 + \frac{h_{ij}P}{N_0})$ where

P is the PSD and $h_{ij} = \frac{1}{d_{ij}^{\alpha}}$ where $\alpha \geq 2$ is the path loss coefficient. Under these assumptions, the following Lemma characterizes the optimal cooperating set for the problem of routing with mutual information accumulation. Its proof is provided in Appendix D.2.

Lemma 5 *The optimal cooperating set for the line network as described above is given by the set of all relay nodes located between the source and the destination.*

To get an idea of the reduction in delay achieved by using mutual information accumulation over traditional routing, consider the line network example above with n nodes placed between s and d at equal distance such that $d_{i,i+1} = 1$ for all i. Also, suppose the transmission capacity on link $i - j$ is given by $C_{ij} = \frac{\gamma P}{d_{ij}^2}$ where $\gamma > 0$ is a constant. Then the capacity of link $s - 1$ is γP, the capacity of link $s - 2$ is $\frac{\gamma P}{4}$, the capacity of link $s - 3$ is $\frac{\gamma P}{9}$, and so on. Define $\theta \triangleq \gamma P$. Then, the minimum delay for routing with mutual information accumulation is given by $\sum_{i=0}^{n} \Delta_i$ where:

$$\Delta_0 = \frac{I_{max}}{C_{s1}} = \frac{I_{max}}{\theta}, \Delta_1 = \frac{I_{max} - \Delta_0 C_{s2}}{C_{12}} = \frac{I_{max} - \Delta_0 \frac{\theta}{4}}{\theta}$$

$$\vdots$$

$$\Delta_n = \frac{I_{max} - \sum_{i=0}^{n-1} \Delta_i C_{i,n+1}}{C_{n,n+1}} = \frac{I_{max} - \sum_{i=0}^{n-1} \Delta_i \frac{\theta}{(n+1-i)^2}}{\theta}$$

For simplicity, let us ignore the contribution of nodes that are more than 3 units away from a receiver. Then, we have:

$$\sum_{i=0}^{n} \Delta_i = \frac{(n+1)I_{max} - \frac{\theta}{4}\sum_{i=0}^{n-1}\Delta_i - \frac{\theta}{9}\sum_{i=0}^{n-2}\Delta_i}{\theta}$$

$$\Rightarrow \sum_{i=0}^{n} \Delta_i = \frac{(n+1)I_{max} + \frac{\theta}{4}\Delta_n + \frac{\theta}{9}(\Delta_n + \Delta_{n-1})}{\theta(1+\frac{1}{4}+\frac{1}{9})} < \frac{(n+1)I_{max} + \frac{\theta}{4}\Delta_0 + \frac{\theta}{9}2\Delta_0}{\theta(1+\frac{1}{4}+\frac{1}{9})}$$

$$= \frac{I_{max}}{\theta}\left(\frac{n+1+\frac{1}{4}+\frac{2}{9}}{1+\frac{1}{4}+\frac{1}{9}}\right)$$

where we used the fact that $\Delta_n, \Delta_{n-1} < \Delta_0$. The minimum delay for traditional routing is simply $(n+1)\Delta_0 = (n+1)\frac{I_{max}}{\theta}$. Thus, for this network, the delay under mutual information accumulation is smaller than that under traditional routing at least by a factor $\frac{n+1+\frac{1}{4}+\frac{2}{9}}{(n+1)(1+\frac{1}{4}+\frac{1}{9})}$ that approaches $\frac{36}{49} = 73\%$ for large n.

5.4 Minimum Energy Routing with Delay Constraint

Next, we consider the second problem of minimizing the sum total energy to transmit a packet from the source to destination using mutual information accumulation subject to a given delay constraint D_{max}. This problem is more challenging than problem (5.1) since in addition to optimizing over the cooperating relay set and the order of transmission, it also involves determining the PSD values to be used for each node. Further, a cooperating relay node may need to transmit at different PSD levels in different stages of the transmission schedule.

5.4.1 Problem Formulation

Consider a transmission strategy (similar to the one discussed in Sec. 5.3.1) that is described by a cooperating relay set $\mathcal{R}$ of size $|\mathcal{R}| = k$ and a decoding order $\mathcal{O}$. Let the terms Δ_j and A_{ij} be defined in a similar fashion. Also, let P_{ij} denote the PSD at which node i transmits in stage j. Then for any transmission strategy $\mathcal{G}$ that uses the subset of relay nodes $\mathcal{R}$ with an ordering $\mathcal{O}$, the minimum sum total energy to transmit a packet from source to destination subject to the delay constraint D_{max} is given by the solution to the following optimization problem:

$$
\begin{aligned}
\text{Minimize:} \quad & \sum_{j=0}^{k}\sum_{i=0}^{j} A_{ij}P_{ij} \\
\text{Subject to:} \quad & \sum_{j=0}^{k} \Delta_j \leq D_{max} \\
& \sum_{i=0}^{m-1}\sum_{j=0}^{m-1} A_{ij}C_{im}(P_{ij}) \geq I_{max} \; \forall m \in \{1, \ldots, k+1\} \\
& \sum_{i=0}^{j} A_{ij} \leq \Delta_j \; \forall j \in \{0, 1, 2, \ldots, k\} \\
& A_{ij}, P_{ij} \geq 0 \; \forall i \in \{0, 1, 2, \ldots, k\}, j \in \{0, 1, \ldots, k\} \\
& A_{ij} = 0, P_{ij} = 0 \; \forall i > j, \; \Delta_j \geq 0 \; \forall j \in \{0, 1, 2, \ldots, k\}
\end{aligned} \tag{5.9}
$$

where the first constraint represents requirement that the total delay must not exceed D_{max}. The second constraint captures the requirement that node m in the ordering must accumulate at least I_{max} amount of mutual information by the end of stage $m-1$ using all transmissions in all stages up to stage $m-1$. In the second constraint, $C_{im}(P_{ij})$ denotes the transmission capacity of link $i - m$ in stage j and it is a function of P_{ij}, the PSD of node i in stage j. Note that (5.9) is not a linear program in general, since the $C_{im}(P_{ij})$ may be non-linear in P_{ij}. Also note that the solution to (5.9) may result in a decoding order that is different from $\mathcal{O}$ in which case that decoding order is infeasible.

5.4.2 Characterizing the Optimal Solution of (5.9)

Let $\mathcal{R}_{opt}$ denote the subset of relay nodes that take part in the routing process in the optimal solution. Let $k = |\mathcal{R}_{opt}|$ be the size of this set. Also, let $\mathcal{O}_{opt}$ be the optimal ordering. Note that, by definition, each node in $\mathcal{R}_{opt}$ transmits for a non-zero duration (else, we can remove it from the set without affecting the sum total energy). Finally, let P_{ij}^{opt} denote the optimal PSD used by node i in stage j. Then, similar to Theorem 6, we have the following:

Theorem 7 *Under the optimal solution to the minimum sum total energy subject to delay constraint problem* (5.9)*, in each stage j, it is optimal for only one node to transmit, and that node is node j.*

Proof 6 *The proof is similar to the proof of Theorem 6 and is omitted for brevity.*

Although Theorem 7 simplifies the optimization problem (5.9), it cannot be solved using the greedy transmission strategy applied over all subsets as discussed in Sec. 5.3.3. This is because the transmission order generated by the greedy strategy depends on the power levels used. For general non-linear rate-power functions, different power levels can give rise to different decoding orders for the same relay set under the greedy strategy (see Appendix D.3 for an example). Thus, solving (5.9) may involve searching over all possible orderings of all possible subsets. However, for the special, yet important case of *linear* rate-power functions, this problem can be simplified considerably. A linear rate-power function is a good approximation for the low SNR regime. For example, in sensor networks where bandwidth is plentiful and power levels are small, it is reasonable to assume that the nodes operate in the low SNR regime. In the following, we will assume that the transmission capacity $C_{ij}(P_i)$ on link $i-j$ is given by $C_{ij}(P_{ij}) = \gamma P_i h_{ij}$ (in units of bits/sec/Hz) where γ is a constant and P_i is the PSD of node i. Then, we have the following:

Theorem 8 *For linear rate-power functions, the decoding order of nodes in the optimal set $\mathcal{R}_{opt}$ under the greedy transmission strategy is the same for all non-zero power allocations. Further, the sum total power required to transmit a packet from the source to the destination is the same for all non-zero power allocations.*

Proof 7 *We prove by induction. Consider any non-zero power allocation used by the nodes in $\mathcal{R}_{opt}$. The source is the first node to transmit. Let it be indexed by 0. Also, suppose the source uses PSD $P_0 > 0$. Under the greedy transmission strategy, the source continues to transmit until any node can decode the packet. This node is the one that minimizes $\Delta_0 = \frac{I_{max}}{C_{0i}(P_0)} = \frac{I_{max}}{\gamma P_0 h_{0i}}$ over all $i \in \mathcal{R}_{opt}$, which is the time to decode the packet. Clearly, this node is the same for all $P_0 > 0$. Let it be indexed by 1. Also, we have that:*

$$\Delta_0 = \frac{I_{max}}{\gamma P_0 h_{01}} \Rightarrow \Delta_0 P_0 = \frac{I_{max}}{\gamma h_{01}}$$

which shows that the total power used in stage 0 *is independent of* P_0*. Next, let the PSD of node* 1 *be* P_1*. Then, in stage* 1 *under the greedy transmission strategy, node* 1 *transmits until any node that does not have the packet yet can decode it. This node is the one that minimizes over all* $i \in \mathcal{R}_{opt} \setminus \{1\}$*:*

$$\frac{I_{max} - \Delta_0 C_{0i}(P_0)}{C_{1i}} = \frac{I_{max} - \Delta_0 \gamma P_0 h_{0i}}{\gamma P_1 h_{1i}} = \frac{I_{max}(1 - \frac{h_{0i}}{h_{01}})}{\gamma P_1 h_{1i}}$$

Clearly, this node is the same for all $P_1 > 0$*. Let it be indexed by* 2*. Also, we have that:*

$$\Delta_1 = \frac{I_{max}(1 - \frac{h_{02}}{h_{01}})}{\gamma P_1 h_{12}} \Rightarrow \Delta_1 P_1 = \frac{I_{max}(1 - \frac{h_{02})}{\gamma h_{01}}}{\gamma h_{12}}$$

which shows that the total power used in stage 1 *is independent of* P_0 *and* P_1*.*

Now suppose this holds for all stages $\{0, 1, 2, \ldots, j-1\}$ *where* $j - 1 < k$*. We show that it also holds for stage* j*. Let the PSD of node* j *be* P_j*. Under the greedy strategy, node* j *continues to transmit in stage* j *until any node that does not have the packet yet can decode it. This node is the one that minimizes over all* $i \in \mathcal{R}_{opt} \setminus \{1, 2, \ldots, j\}$*:*

$$\frac{I_{max} - \sum_{m=0}^{j-1} \Delta_m C_{mi}(P_m)}{C_{ji}(P_j)} = \frac{I_{max} - \gamma \sum_{m=0}^{j-1} \Delta_m P_m h_{mi}}{\gamma P_j h_{ji}}$$

From the induction hypothesis, we know that each of the terms $\Delta_m P_m$ *for all* $m \in \{0, 1, \ldots, j-1\}$ *is independent of the power levels* P_m*. Thus, we have that the node* *that minimizes the expression above is the same for all $P_j > 0$. Further, the total power used in stage j is given by*

$$\Delta_j P_j = \frac{I_{max} - \gamma \sum_{m=0}^{j-1} \Delta_m P_m h_{mi}}{\gamma h_{ji}}$$

which is independent of $P_0, P_1, \ldots, P_m$. This proves the Theorem.

5.4.3 A Greedy Algorithm

Theorem 8 suggests a simple method for computing the optimal solution to (5.9) when the rate-power function is linear. Specifically, we start by setting all PSD levels to the same value, say some $P > 0$. From Theorem 8, we know that the sum total power required to transmit a packet from the source to the destination is the same for all non-zero power allocations. Then, solving (5.9) is equivalent to solving the minimum delay problem (5.1) with given power levels, except the delay constraint. This can be done using the greedy strategy described in Sec. 5.3.3. If the solution obtained satisfies the delay constraint D_{max}, then we are done. Else, suppose we get a delay $D > D_{max}$. Then, we can scale up the power level P by a factor $\frac{D}{D_{max}}$ and scale down the duration of each stage Δ_j by the same factor. This ensures that the delay constraint is met while the sum total power used remains the same.

5.5 Minimum Delay Broadcast

Next, we consider the problem of minimum delay broadcast for the network model described in Sec. 5.2. In this problem, starting with the source node, the goal is to deliver the packet to all nodes in the network in minimum time with mutual information accumulation. We assume that there are n nodes in the network other than the source. Similar problems have been considered in [ACGW04, MY04a, SMS07] which focus on energy accumulation and where the goal is to broadcast the packet to all nodes using minimum sum total energy.

5.5.1 Timeslot and Transmission Structure

For the minimum delay broadcast problem, the transmission strategy and resulting time timeslot structure under a general policy is similar to the one discussed for the minimum delay routing problem in Sec. 5.3.1. Specifically, let $\mathcal{O}$ be the ordering of the n nodes that represents the sequence in which they successfully decode the packet under a given strategy. Without loss of generality, let the nodes in the ordering $\mathcal{O}$ be indexed as $1, 2, 3, \ldots, n$. Also, let the source s be indexed as 0. Initially, only the source has the packet. Let t_0 be the time when it starts its transmission and let $t_1, t_2, \ldots, t_n$ denote the times when nodes $1, 2, \ldots, n$ in the ordering $\mathcal{O}$ accumulate enough mutual information to decode the packet. We say that the transmission occurs over n stages, where stage $j, j \in \{0, 1, 2, \ldots, n-1\}$ represents the interval $[t_j, t_{j+1}]$. Note that in any stage j, the first j nodes in the ordering $\mathcal{O}$ and the source have the fully decoded packet. Thus, any subset of these nodes (including potentially all of them) may transmit during this stage. For each j, define the duration of stage j as $\Delta_j = t_{j+1} - t_j$. Also, let A_{ij} denote the transmission duration for node i in stage j. As before, we have that $A_{ij} = 0$ if $i > j$, else $A_{ij} \geq 0$. The total time to deliver the packet to all the n nodes is given by $T_{tot} = t_n - t_0 = \sum_{j=0}^{n-1} \Delta_j$.

5.5.2 Problem Formulation

For any transmission strategy that results in the decoding order $\mathcal{O}$, the minimum delay for broadcast is given by the solution to the following optimization problem:

$$
\begin{aligned}
\text{Minimize:} \quad & T_{tot} = \sum_{j=0}^{n-1} \Delta_j \\
\text{Subject to:} \quad & \sum_{i=0}^{m-1} \sum_{j=0}^{m-1} A_{ij} C_{im} \geq I_{max} \ \forall m \in \{1, 2, \ldots, n\} \\
& \sum_{i=0}^{j} A_{ij} \leq \Delta_j \ \forall j \in \{0, 1, 2, \ldots, n-1\} \\
& A_{ij} \geq 0 \ \forall i \in \{0, 1, 2, \ldots, n-1\}, j \in \{0, 1, 2, \ldots, n-1\} \\
& A_{ij} = 0 \ \forall i > j \\
& \Delta_j \geq 0 \ \forall j \in \{0, 1, 2, \ldots, n-1\}
\end{aligned} \tag{5.10}
$$

This is similar to (5.1) except that the set $\mathcal{R}$ contains all n nodes and that d is not necessarily the last node to decode the packet. Similar to (5.1), the first constraint captures the requirement that node m in the decoding order $\mathcal{O}$ must accumulate at least I_{max} amount of mutual information by the end of stage $m-1$ using transmissions in all stages up to stage $m-1$. The second constraint means that in every stage j, the total transmission time for all nodes that have the fully decoded packet in that stage cannot exceed the length of that stage. We note that the solution to (5.10) may result in a decoding order that is different from $\mathcal{O}$. In that case, the decoding order $\mathcal{O}$ is infeasible.

Similar to (5.1), the above problem is a linear program and thus can be solved efficiently for a given ordering $\mathcal{O}$. This is the approach taken in [MY04a] (with energy accumulation instead of mutual information accumulation, and with the objective of minimizing total energy for broadcast instead of delay) that proposes solving such a linear program for *every possible ordering* of the n nodes, resulting in $n!$ linear programs. In the next section, we show that the above computation can be simplified by making use of a structural property of the optimal solution that is similar to the results of Theorems 6 and 7. This results in a greedy algorithm that does not require solving such linear programs to compute the optimal solution.

5.5.3 Characterizing the Optimal Solution of (5.10)

Let $\mathcal{O}_{opt}$ be the decoding order under the optimal solution. Suppose the the nodes in the ordering are labeled as $\{0, 1, 2, \ldots, n-1, n\}$ with 0 being the source node. Then, similar to Theorems 6 and 7, we have the following:

Theorem 9 *Under the optimal solution to the minimum delay broadcast problem* (5.10), *in each stage j, it is optimal for at most one node to transmit.*

While Theorem 9 states that under the optimal solution, at most one node transmits in each stage j, unlike Theorems 6 and 7, it does not say that this node must be node j. In fact, this node could be any one of the nodes that have the full packet. Let r_j be the node that transmits in stage j. Then, using Theorem 9, we have that $r_j \in \{0, 1, 2, \ldots, j\}$.

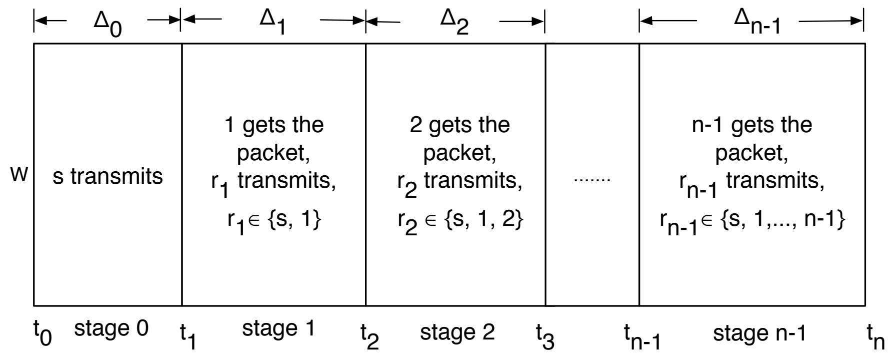

Figure 5.5: Optimal timeslot and transmission structure for minimum delay broadcast. In each stage, at most one node from the set of nodes that have the full packet transmits.

The optimal timeslot structure for the minimum delay broadcast problem is shown in Fig. 5.5. Note that unlike Fig. 5.3, here it is possible for a node to transmit more than once over the course of the broadcast.

This property does not reduce the complexity of finding the optimal solution from $O(n!)$ linear programs to $O(2^n)$. However, as we show in Sec. 5.5.5, it still leads to a *greedy* algorithm to find the optimal solution that does not require solving $n!$ linear programs like [MY04a].

5.5.4 Proof of Theorem 9

The proof is similar to the proof of Theorem 6 and therefore, we only provide a sketch here, highlighting the main differences.

Note that Theorem 9 trivially holds in stage 0 (since only the source has the full packet in this stage). Next, similar to Theorem 6, in the last stage (i.e., stage $(n-1)$), only the node with the *best* link (in terms of transmission capacity) to node n in the ordering $\mathcal{O}_{opt}$ should transmit in order to minimize the total delay. Let this node be labeled r_{n-1}. However, unlike Theorem 6, we cannot claim that this node must be node $n-1$ in the ordering $\mathcal{O}_{opt}$. This is because while r_{n-1} has the best link to n, it does not necessarily have the best links to all those nodes in the decoding order $\mathcal{O}_{opt}$ that come after r_{n-1}. Thus r_{n-1} could be any one of $\{0, 1, 2, \ldots, n-1\}$. This shows that under the optimal solution, in the last stage $(n-1)$, only one node r_{n-1} transmits. Using induction, we can show that in every prior stage $(n-j)$ where $1 < j < n$, at most one node needs to transmit.

Consider the $(n-2)^{th}$ stage. At time t_{n-2}, all nodes except $n-1$ and n have decoded the packet. Let the mutual information state at nodes $n-1$ and n at time t_{n-2} be $I_{n-1}(t_{n-2})$ and $I_n(t_{n-2})$ respectively. Also, suppose in the $(n-2)^{th}$ stage, relay nodes $1, 2, \ldots, n-2$ and the source transmit a fraction $\alpha_1^{n-2}, \alpha_2^{n-2}, \ldots, \alpha_{n-2}^{n-2}$ and α_0^{n-2} of the total duration of stage $(n-2)$, i.e., Δ_{n-2}, respectively. Note that these fractions must add to 1 since it is suboptimal to have any idle time (where no one is transmitting). Then, the optimal solution must solve the following optimization problem:

$$\begin{aligned}
\text{Minimize:} \quad & \Delta_{n-2} + \Delta_{n-1} \\
\text{Subject to:} \quad & I_{n-1}(t_{n-2}) + \Delta_{n-2} \sum_{i=0}^{n-2} \alpha_i^{n-2} C_{i,n-1} \geq I_{max} \\
& I_n(t_{n-2}) + \Delta_{n-2} \sum_{i=0}^{n-2} \alpha_i^{n-2} C_{in} + \Delta_{n-1} C_{r_{n-1},n} \geq I_{max} \\
& 0 \leq \alpha_0^{n-2}, \alpha_1^{n-2}, \ldots, \alpha_{n-2}^{n-2} \leq 1 \\
& \sum_{i=0}^{n-2} \alpha_i^{n-2} = 1 \\
& \Delta_{n-2} \geq 0, \Delta_{n-1} \geq 0
\end{aligned} \tag{5.11}$$

Here, the first constraint states that node $n-1$ must accumulate at least I_{max} bits of mutual information by the end of stage $(n-2)$. The second constraint states that node n must accumulate at least I_{max} bits of mutual information by the end of stage $(n-1)$. Note that in the last term of the left hand side of the second constraint, we have used the fact that only node r_{n-1} transmits during stage $(n-1)$.

It is easy to see that under the optimal solution, the first and second constraints must be met with equality. This simply follows from the definition of the beginning of any stage j as the time when node j has just decoded the packet. Next, let $\beta_i = \Delta_{n-2}\alpha_i^{n-2}$ for all $i \in \{0, 1, 2, \dots, n-2\}$. Since $\sum_{i=0}^{n-2} \alpha_i^{n-2} = 1$, we have that $\sum_{i=0}^{n-2} \beta_i = \Delta_{n-2}$ and (5.11) is equivalent to:

$$\begin{aligned}
\text{Minimize:} \quad & \sum_{i=0}^{n-2} \beta_i + \Delta_{n-1} \\
\text{Subject to:} \quad & I_{n-1}(t_{n-2}) + \sum_{i=0}^{n-2} \beta_i C_{i,n-1} = I_{max} \\
& I_n(t_{n-2}) + \sum_{i=0}^{n-2} \beta_i C_{in} + \Delta_{n-1} C_{r_{n-1},n} = I_{max} \\
& \Delta_{n-1} \geq 0, \beta_i \geq 0 \qquad \forall i \in \{0, 1, 2, \dots, n-2\}
\end{aligned} \tag{5.12}$$

Note that problems (5.11) and (5.12) are equivalent because we can transform (5.12) to the original problem by using the relations $\Delta_{n-2} = \sum_{i=0}^{n-2} \beta_i$ and $\alpha_i^{n-2} = \frac{\beta_i}{\Delta_{n-2}}$. In the degenerate case where $\Delta_{n-2} = 0$, we have that no node transmits in stage $(n-2)$, so that Theorem 9 holds.

Using similar arguments as in Theorem 6, it can be shown that when $\Delta_{n-2} > 0$, then in the optimal solution exactly one of the fractions $\alpha_0^{n-2}, \alpha_1^{n-2}, \dots, \alpha_{n-2}^{n-2}$ is equal to 1 and rest must be 0. This implies that only one node transmits in this stage. Combining with the case where $\Delta_{n-2} = 0$, we have that at most one node transmits in stage $(n-2)$. We label this node as r_{n-2}. Note that r_{n-2} could be any one of $\{0, 1, 2, \ldots, n-2\}$.

Using induction, it can be shown that in every stage $(n-j), 2 < j < n$, at most one node labeled r_{n-j} transmits. Further, r_{n-j} could be any one of $\{0, 1, 2, \ldots, n-j\}$. This proves the Theorem.

5.5.5 A Greedy Algorithm

Theorem 9 can be used to construct the following greedy algorithm for computing the optimal solution to (5.10). The algorithm operates over n stages. In each stage j, $0 \leq j \leq n-1$, the algorithm performs $(j+1)!$ *separate* runs as discussed below. Let $\mathcal{S}_{ij}$ denote the set of nodes that have the full packet at the end of the i^{th} run of stage j. Then, each run in stage $j+1$ corresponds to selecting one transmitter from each $\mathcal{S}_{ij}$ and having that node transmit until a new node decodes the packet. Thus, the number of nodes with the full packet increases by one at the end of each run. We will show that the size of $\mathcal{S}_{ij}$ is equal to $\|\mathcal{S}_{ij}\| = j+2$ for all i, j. Further, there are $(j+1)!$ distinct such sets. Thus, the total number of runs in stage $j+1$ becomes $(j+2) \times (j+1)! = (j+2)!$.

To see this, note that we start at stage 0 with only the source having the full packet and perform only one run. At the end of this stage, suppose node 1 has the packet. Thus, $\mathcal{S}_{10} = \{s, 1\}$ and has size $0+2 = 2$. In next stage (i.e., stage 1), we perform $2! = 2$ separate runs as follows. In the first run, s is chosen as the transmitter for stage 1 and continues to transmit until another node (say x) gets the packet. This yields $\mathcal{S}_{11} = \{s, 1, x\}$. In the second run, 1 is chosen as the transmitter for stage 1 and continues to transmit until another node (say y) gets the packet. This yields $\mathcal{S}_{21} = \{s, 1, y\}$. Thus, at the end of stage 1, we have $2! = 2$ sets, $\mathcal{S}_{11}$ and $\mathcal{S}_{21}$, of size $1 + 2 = 3$ each.

This procedure is repeated in stage 2 resulting in 3 runs starting with $\mathcal{S}_{11}$ and 3 runs starting with $\mathcal{S}_{21}$. Thus, in stage 2, the algorithm performs $(2+1)! = 6$ runs and yields $3! = 6$ sets, $\mathcal{S}_{13}, \mathcal{S}_{23}, \ldots, \mathcal{S}_{63}$, each of size $2 + 2 = 4$, at the end of stage 2. In the same way, it can be shown that in stage j, the algorithm starts with $j!$ sets of size $j + 1$ each, performs $(j + 1)!$ runs and results in $(j + 1)!$ sets, each of size $j + 2$.

The algorithm terminates after stage $(n - 1)$ where it performs $n!$ runs and when all nodes decode the packet. The optimal solution is obtained by picking the sequence of transmitting nodes that yields the minimum delay.

It can be seen that the complexity of this algorithm is $O(n!)$ Essentially, this algorithm performs an exhaustive search over all possible feasible decoding orderings. This corresponds to searching over all possible values of $r_j \in \{s, 1, 2, \ldots, j\}$ in every stage j (See Fig. 5.5). However, unlike [MY04a], it does not require solving any linear programs.

5.6 Distributed Heuristics and Simulations

The greedy algorithm presented in Sec. 5.3.3 to compute the optimal solution to problem (5.1) has an exponential computational complexity and is centralized. In this section, we present two simple heuristics that can be implemented in polynomial time and in a distributed fashion. We compare the performance of these heuristics with the optimal solution on general network topologies. We also show the performance of the traditional minimum delay route that does not use mutual information accumulation.

Heuristic 1: Here, first the traditional minimum delay route is computed using, say, Dijkstra's shortest path algorithm on the weighted graph (where the weight w_{ij} of link $i - j$ is defined as the time required to deliver a packet from i to j, i.e., $w_{ij} = \frac{I_{max}}{C_{ij}}$). Let $\mathcal{M}$ denote the set of relay nodes that form this minimum delay shortest path. Then the greedy algorithm as described in Sec. 5.3.3 is applied on the set of nodes in $\mathcal{M}$. Note that we are not searching over all subsets of $\mathcal{M}$. It may be possible to get further gains by searching over all subsets of $\mathcal{M}$, but the worst case complexity of doing so would again be exponential. Our goal here is to develop polynomial time algorithms. Thus, the complexity of this heuristic is same as that of any shortest path algorithm, i.e., $O(|\mathcal{M}|^2)$.

Heuristic 2: Here, we start with $\mathcal{M}$ as the initial cooperative set. Then, while applying the greedy algorithm of Sec. 5.3.3, if other nodes that are not in $\mathcal{M}$ happen to decode the packet before the next node (where the next node is defined as that node in $\mathcal{M}$ that would decode the packet if the current transmitter continued its transmission), then these nodes are added to the cooperative set if they have a better channel to the next node than the current transmitter. The intuition behind this heuristic is that while $\mathcal{M}$ is expected to be a good cooperative set, this allows the algorithm to explore more nodes and potentially improve over Heuristic 1.

5.6.1 Simulation Results

In our simulations, we consider a network of a source, destination, and n relay nodes located in a 10×10 area. The location of source $(1.0, 2.0)$ and destination $(8.0, 8.0)$ is fixed while the locations of the other nodes are chosen uniformly at random. The link gain h_{ij} between any two nodes i and j is chosen from a Rayleigh distribution with mean 1.

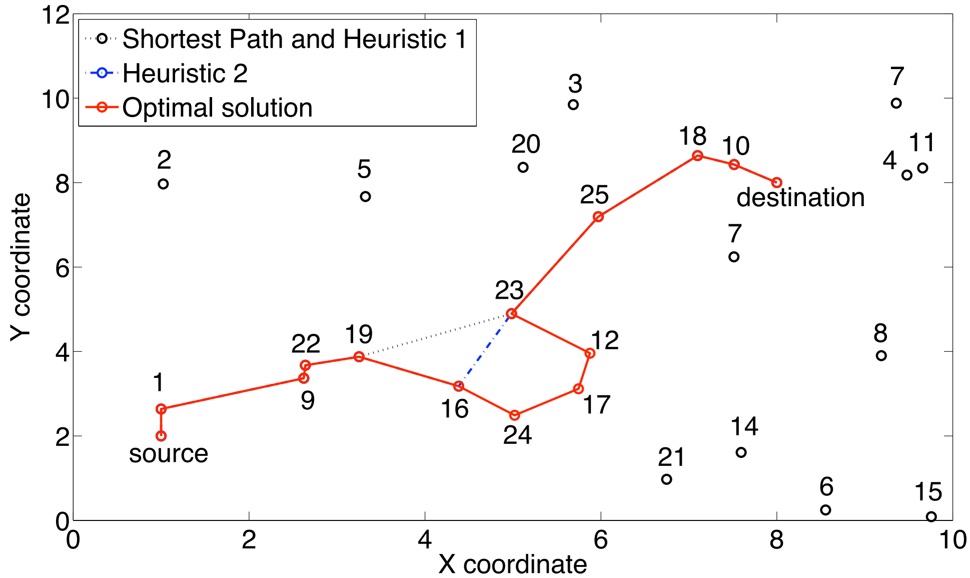

Figure 5.6: A 25 node network where the routes for traditional minimum delay, Heuristics 1 and 2, and optimal mutual information accumulation are shown.

For simplicity, all nodes have the same normalized PSD of 1. Also, $W = 1$ and $I_{max} = 1$. The transmission capacity of link $i - j$ is assumed to be $C_{ij} = \log_2\left(1 + \frac{h_{ij}}{d_{ij}^{\alpha}}\right)$ where d_{ij} is the distance between nodes i and j and α is the path loss exponent. We choose $\alpha = 3$ for all simulations.

In the first simulation, $n = 25$ and the network topology is fixed as shown in Fig. 5.6. We then compute the traditional minimum delay route and the optimal solution for routing with mutual information accumulation using the greedy algorithm of Sec. 5.3.3. We also implement Heuristics 1 and 2 on this network. Fig. 5.6 shows the results. It is seen that the traditional minimum delay route is given by $[s, 1, 9, 22, 19, 23, 25, 18, 10, d]$ while the optimal mutual information accumulation route (according to the decoding order) is given by $[s, 1, 9, 22, 19, 16, 24, 17, 12, 23, 25, 18, 10, d]$. The decoding order of nodes under Heuristic 1 is same as that under the traditional minimum delay route while that

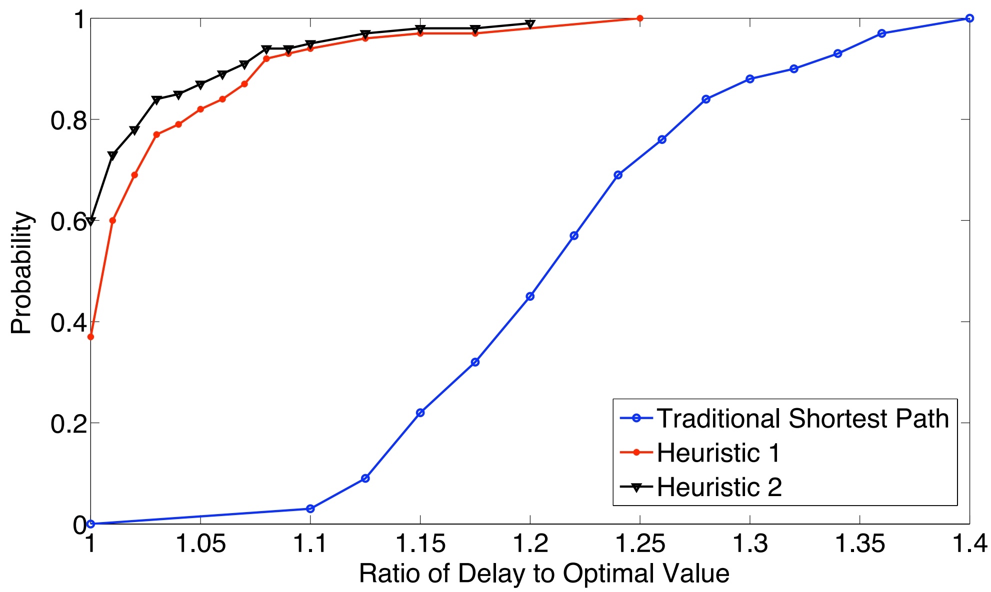

Figure 5.7: The CDF of the ratio of the minimum delay under the two heuristics and the traditional shortest path to the minimum delay under the optimal mutual information accumulation solution.

under Heuristic 2 is given by $[s, 1, 9, 22, 19, 16, 23, 25, 18, 10, d]$. The total delay under traditional minimum delay routing, Heuristic 1, Heuristic 2, and optimal mutual information accumulation routing was found to be $29.84, 23.73, 22.99$ and 22.19 seconds respectively.

This example demonstrates that the optimal route under mutual information accumulation can be quite different from the traditional minimum delay path. It is also interesting to note that the set of nodes in $\mathcal{M}$ is a subset of the cooperative relay set in this example. However, this does not hold in general. We also note that the delay under both Heuristics 1 and 2 is close to the optimal value. Finally, while Heuristic 1 only uses the nodes in $\mathcal{M}$, Heuristic 2 explores more and ends up using node 16 as well.

In the second simulation, we choose $n = 20$. The source and destination locations are fixed as before but the locations of the relay nodes are varied randomly over 100 instances. For each topology instance, we compute the minimum delay obtained by these

4 algorithms. In Fig. 5.7, we plot the cumulative distribution function (CDF) of the ratio of the minimum delay under the two heuristics and the traditional shortest path to the minimum delay under the optimal mutual information accumulation solution. From this, it can be seen that both Heuristic 1 and 2 perform quite well over general network topologies. In fact, they are able to achieve the optimal performance 40% and 60% of the time respectively. Further, they are within 10% of the optimal at least 90% of the time and within 15% of the optimal at least 99% of the time. Also, Heuristic 2 is seen to outperform Heuristic 1 in general. Finally, the average delay gain in routing with mutual information accumulation over traditional shortest path was found to be 77%.

5.7 Chapter Summary

In this chapter, we considered three problems involving optimal routing and scheduling over a multi-hop wireless network using mutual information accumulation. We formulated the general problems as combinatorial optimization problems and then made use of several structural properties to simplify their solutions and derive optimal greedy algorithms. A key feature of these algorithms is that unlike prior works on these problems, they do not require solving any linear programs to compute the optimal solution. While these greedy algorithms still have exponential complexity, they are significantly simpler than prior schemes and allows us to compute the optimal solution as a benchmark. We also proposed two simple and practical heuristics that exhibit very good performance when compared to the optimal solution.

In this work, our focus has been on the “one-shot” problem of optimal routing/broadcasting of a single packet in a static wireless network. An immediate future work involves investigating the throughput region associated with both single and multiple flows in a time-varying network when mutual information accumulation is used.

Chapter 6

Conclusions

In this thesis, we studied four problems on optimal resource allocation and cross-layer control in cognitive and cooperative wireless networks with time-varying channels. The first three problems investigated different models and capabilities associated with cognition and cooperation in such networks. We first considered the dynamic spectrum access model in a cognitive radio network with primary and secondary users where the primary users are licensed owners of spectrum while the secondary users do no have any such licensed spectrum. The primary users are oblivious to the presence of the secondary users and transmit on their licensed channels whenever they have data to send. The secondary users have imperfect knowledge about the primary users' spectrum usage and must meet a constraint on the maximum time-average rate of collisions for each primary user while seeking transmission opportunities on idle primary channels. In the second problem, we considered a fully cooperative wireless network where the nodes use relay-based cooperative communication to improve each other's transmission rates. Different from the first problem, this can model a cognitive network where there is no such differentiation between primary and secondary users. In the third problem, we considered a cognitive radio model where the primary users are aware of the presence of the secondary users but have strictly higher priority in accessing their channels. In this scenario, the secondary users can use their resources to improve the transmission rate of the primary user. This can create more opportunities for the secondary users to transmit their own data on the primary channels.

In all of these problems, our goal was to design optimal control algorithms that maximize time-average network utilities (such as throughput) subject to time-average constraints (such as power, reliability, etc.). To this end, we made use of the technique of Lyapunov optimization to design online control algorithms for these problems. The three problems we studied are structurally different from each other. Therefore, the traditional Lyapunov optimization technique had to be adjusted appropriately in order to solve them. In the first problem, we used a greedy drift-plus-penalty minimizing algorithm over every slot. In the second problem, the drift-plus-penalty was minimized over every frame (where each frame consists of two stages). Finally, in the third problem, we used a drift-plus-penalty-ratio minimization approach. Here, the ratio of the expected total drift-plus-penalty over the expected length of a frame is minimized every frame. In all three cases, the resulting algorithms that we developed are *greedy* and *myopic* in nature. They can operate without requiring *any* knowledge of the statistical description of network dynamics (such as fading channels, node mobility, and random packet arrivals) and are provably optimal.

Finally, in the fourth problem, we investigated optimal routing and scheduling in static wireless networks with rateless codes. Rateless codes allow each node of the network to accumulate mutual information with every packet transmission. This enables a significant performance gain over conventional shortest path routing. Further, it also outperforms cooperative communication techniques that are based on energy accumulation. However, it requires complex and combinatorial networking decisions concerning which nodes participate in transmission, and which decode ordering to use. We formulated the general problems as combinatorial optimization problems and identified several structural properties of the optimal solutions. This enabled us to derive optimal greedy algorithms to solve these problems.

Bibliography

[ACGW04] M. Agarwal, J. H. Cho, L. Gao, and J. Wu. Energy efficient broadcast in wireless ad hoc networks with hitch-hiking. In *Proc. IEEE INFOCOM*, pages 2096–2107, March 2004.

[Alt99] E. Altman. *Constrained Markov Decision Processes.* Chapman and Hall/CRC Press, Boca Raton, FL, 1999.

[ALVM06] I. F. Akyildiz, W.-Y. Lee, M. C. Vuran, and S. Mohanty. Next generation/dynamic spectrum access/cognitive radio wireless networks: A survey. *Computer Networks*, 50:2127–2159, Sept. 2006.

[Ber07] D. P. Bertsekas. *Dynamic Programming and Optimal Control, vols. 1 and 2.* Athena Scientific, Belmont, MA, 2007.

[BK11] M. Baghaie and B. Krishnamachari. Delay constrained minimum energy broadcast in cooperative wireless netorks. In *Proc. IEEE INFOCOM*, April 2011.

[BLM02] J.W. Byers, M. Luby, and M. Mitzenmacher. A digital fountain approach to asynchronous reliable multicast. *IEEE Journal on Selected Areas in Communications*, 20(8):1528–1540, Oct. 2002.

[BT96] D. P. Bertsekas and J. N. Tsitsiklis. *Neuro-Dynamic Programming.* Athena Scientific, Belmont, MA, 1996.

[Bud07] M. M. Buddhikot. Understanding dynamic spectrum access: Models, taxonomy and challenges. In *2nd IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks*, pages 649–663, April 2007.

[BV04] S. Boyd and L. Vandenberghe. *Convex Optimization.* Cambridge University Press, 2004.

[BZG07] S. Borade, L. Zheng, and R. Gallager. Amplify-and-forward in wireless relay networks: Rate, diversity, and network size. *IEEE Transactions on Information Theory*, 53(10):3302–3318, Oct. 2007.

[Car78] A. Carleial. Interference channels. *IEEE Transactions on Information Theory*, 24(1):60–70, Jan. 1978.

[CG79] T. Cover and A. E. Gamal. Capacity theorems for the relay channel. *IEEE Transactions on Information Theory*, 25(5):572–584, Sep. 1979.

[CJL+05] J. Chen, L. Jia, X. Liu, G. Noubir, and R. Sundaram. Minimum energy accumulative routing in wireless networks. In *Proc. IEEE INFOCOM*, pages 1875–1886, March 2005.

[CKLS08] P. Chaporkar, K. Kar, X. Luo, and S. Sarkar. Throughput and fairness guarantees through maximal scheduling in wireless networks. *IEEE Transactions on Information Theory*, 54(2):572–594, Feb. 2008.

[CM07] J. Castura and Y. Mao. Rateless coding and relay networks. *IEEE Signal Processing Magazine*, 24(5):27–35, Sept. 2007.

[CSY08] M. Chen, S. Serbetli, and A. Yener. Distributed power allocation strategies for parallel relay networks. *IEEE Transactions on Wireless Communications*, 7(2):552–561, Feb. 2008.

[CT91] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. John Wiley & Sons, Inc., New York, 1991.

[CT01] G. Caire and D. Tuninetti. The throughput of hybrid-arq protocols for the gaussian collision channel. *IEEE Transactions on Information Theory*, 47(5):1971–1988, July 2001.

[CTB99] G. Caire, G. Taricco, and E. Biglieri. Optimum power control over fading channels. *IEEE Transactions on Information Theory*, 45(5):1468–1489, July 1999.

[CZ05] L. Cao and H. Zheng. Distributed spectrum allocation via local bargaining. In *Proc. IEEE SECON*, pages 475–486, Sept. 2005.

[CZS08] Y. Chen, Q. Zhao, and A. Swami. Joint design and separation principle for opportunistic spectrum access in the presence of sensing errors. *IEEE Transactions on Information Theory*, 54(5):2053–2071, May 2008.

[DFGV10] H. Dubois-Ferriere, M. Grossglauser, and M. Vetterli. Valuable detours: Least-cost anypath routing. *IEEE/ACM Transactions on Networking*, PP(99):1, 2010.

[DGG10] M. Dehghan, M. Ghaderi, and D. L. Goeckel. Cooperative diversity routing in wireless networks. In *Proc. of the 8th International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt), 2010*, pages 31–39, June 2010.

[DLMY08] S. C. Draper, L. Liu, A. F. Molisch, and J. S. Yedidia. Routing in cooperative wireless networks with mutual-information accumulation. In *Proc. IEEE International Conference on Communications*, pages 4272–4277, May 2008.

[DSM09] S. Deb, V. Srinivasan, and R. Maheshwari. Dynamic spectrum access in dtv whitespaces: Design rules, architecture and algorithms. In *Proceedings of the 15th annual international conference on Mobile computing and networking*, MobiCom '09, pages 1–12, New York, NY, USA, 2009. ACM.

[Gal96] R. Gallager. *Discrete Stochastic Processes.* Kluwer Academic Publishers, Boston, MA, 1996.

[GBA10] G. Gur, S. Bayhan, and F. Alagoz. Cognitive femtocell networks: An overlay architecture for localized dynamic spectrum access. *IEEE Wireless Communications*, 17(4):62–70, Aug. 2010.

[GE07] D. Gunduz and E. Erkip. Opportunistic cooperation by dynamic resource allocation. *IEEE Transactions on Wireless Communications*, 6(4):1446–1454, April 2007.

[GJMS09] A. Goldsmith, S. A. Jafar, I. Maric, and S. Srinivasa. Breaking spectrum gridlock with cognitive radios: An information theoretic perspective. *Proceedings of the IEEE*, 97(5):894–914, May 2009.

[GNT06] L. Georgiadis, M. J. Neely, and L. Tassiulas. Resource allocation and cross-layer control in wireless networks. *Foundations and Trends in Networking*, 1:1–144, April 2006.

[GV05] M. Gastpar and M. Vetterli. On the capacity of large gaussian relay networks. *IEEE Transactions on Information Theory*, 51(3):765–779, March 2005.

[HA04] M. O. Hasna and M.-S. Alouini. Optimal power allocation for relayed transmissions over rayleigh-fading channels. *IEEE Transactions on Wireless Communications*, 3(6):1999–2004, Nov. 2004.

[Hay05] S. Haykin. Cognitive radio: Brain-empowered wireless communications. *IEEE Journal on Selected Areas in Communications*, 23(2):201–220, Feb. 2005.

[HHCK07] Y.-W. Hong, W.-J. Huang, F.-H. Chiu, and C.-C. J. Kuo. Cooperative communications in resource-constrained wireless networks. *IEEE Signal Processing Magazine*, 24(3):47–57, May 2007.

[HK81] T. Han and K. Kobayashi. A new achievable rate region for the interference channel. *IEEE Transactions on Information Theory*, 27(1):49–60, Jan. 1981.

[HLD08] S. Huang, X. Liu, and Z. Ding. Opportunistic spectrum access in cognitive radio networks. In *Proc. IEEE INFOCOM*, pages 1427–1435, April 2008.

[HMZ05] A. Host-Madsen and J. Zhang. Capacity bounds and power allocation for wireless relay channels. *IEEE Transactions on Information Theory*, 51(6):2020–2040, June 2005.

[HN09] L. Huang and M. J. Neely. Delay reduction via lagrange multipliers in stochastic network optimization. In *Proc. of the 7th International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt), 2009*, pages 1–10, June 2009.

[HSS07] Y. T. Hou, Y. Shi, and H. D. Sherali. Optimal spectrum sharing for multi-hop software defined radio networks. In *Proc. IEEE INFOCOM*, pages 1–9, May 2007.

[HT98] S. V. Hanly and D. N. C. Tse. Multiaccess fading channels-part ii: Delay-limited capacities. *IEEE Transactions on Information Theory*, 44(7):2816–2831, Nov. 1998.

[JL10] J. Jin and B. Li. Cooperative resource management in cognitive wimax with femto cells. In *Proc. IEEE INFOCOM*, pages 1–9, March 2010.

[KAMZ07] A. E. Khandani, J. Abounadi, E. Modiano, and L. Zheng. Cooperative routing in static wireless networks. *IEEE Transactions on Communications*, 55(11):2185–2192, Nov. 2007.

[KGG05] G. Kramer, M. Gastpar, and P. Gupta. Cooperative strategies and capacity theorems for relay networks. *IEEE Transactions on Information Theory*, 51(9):3037–3063, Sept. 2005.

[KLTM09] I. Krikidis, J. N. Laneman, J. S. Thompson, and S. McLaughlin. Protocol design and throughput analysis for multi-user cognitive cooperative systems. *IEEE Transactions on Wireless Communications*, 8:4740–4751, Sept. 2009.

[KMY06] G. Kramer, I. Marić, and R. D. Yates. Cooperative communications. *Foundations and Trends in Networking*, 1:271–425, August 2006.

[KS10] M. H. R. Khouzani and S. Sarkar. Economy of spectrum access in time varying multichannel networks. *IEEE Transactions on Mobile Computing*, 9(10):1361–1376, Oct. 2010.

[LDFK09] R. Laufer, H. Dubois-Ferriere, and L. Kleinrock. Multirate anypath routing in wireless mesh networks. In *Proc. IEEE INFOCOM*, pages 37–45, April 2009.

[LKL10] X. Liu, B. Krishnamachari, and H. Liu. Channel selection in multi-channel opportunistic spectrum access networks with perfect sensing. In *2010 IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks*, pages 1–8, April 2010.

[LL09] X. Liu and T. J. Lim. Fountain codes over fading relay channels. *IEEE Transactions on Wireless Communications*, 8(6):3278–3287, June 2009.

[LLS10] M. Lotfinezhad, B. Liang, and E. S. Sousa. Optimal control of constrained cognitive radio networks with dynamic population size. In *Proc. IEEE INFOCOM*, pages 1–9, March 2010.

[LMZ10] M. Levorato, U. Mitra, and M. Zorzi. Cognitive interference management in retransmission-based wireless networks. *arXiv Technical Report: arXiv:1004.0542v1*, April 2010.

[LN10] C.-P. Li and M. J. Neely. Network utility maximization over partially observable markovian channels. *Arxiv Technical Report: arXiv:1008.3421v1*, August 2010.

[LS06] X. Lin and N. B. Shroff. The impact of imperfect scheduling on cross-layer congestion control in wireless networks. *IEEE/ACM Transactions on Networking*, 14(2):302–315, April 2006.

[LT06] C. Lott and D. Teneketzis. Stochastic routing in ad-hoc networks. *IEEE Transactions on Automatic Control*, 51(1):52–70, Jan. 2006.

[LTW04] J. N. Laneman, D. N. C. Tse, and G. W. Wornell. Cooperative diversity in wireless networks: Efficient protocols and outage behavior. *IEEE Transactions on Information Theory*, 50(12):3062–3080, Dec. 2004.

[Lub02] M. Luby. Lt codes. In *Proc. IEEE Symposium on Foundations of Computer Science*, pages 271–280, 2002.

[LW03] J. N. Laneman and G. W. Wornell. Distributed space-time coded protocols for exploiting cooperative diversity in wireless networks. *IEEE Transactions on Information Theory*, 49(10):2415–2425, Oct. 2003.

[MBM07] R. Mudumbai, G. Barriac, and U. Madhow. On the feasibility of distributed beamforming in wireless networks. *IEEE Transactions on Wireless Communications*, 6(5):1754–1763, May 2007.

[Mey08] S. Meyn. *Control Techniques for Complex Networks*. Cambridge University Press, 2008.

[Mit00] J. Mitola. *Cognitive radio: An integrated agent architecture for software defined radio*. PhD thesis, KTH, Stockholm, Sweden, 2000.

[MM99] J. Mitola and G. Q. Maguire. Cognitive radio: Making software radios more personal. *IEEE Personal Communications*, 6(4):13–18, Aug. 1999.

[MMYZ07] A. F. Molisch, N. B. Mehta, J. S. Yedidia, and J. Zhang. Performance of fountain codes in collaborative relay networks. *IEEE Transactions on Wireless Communications*, 6(11):4108–4119, Nov. 2007.

[MY04a] I. Maric and R. D. Yates. Cooperative multihop broadcast for wireless networks. *IEEE Journal on Selected Areas in Communications*, 22(6):1080–1088, Aug. 2004.

[MY04b] I. Maric and R. D. Yates. Forwarding strategies for gaussian parallel-relay networks. In *Proc. International Symposium on Information Theory*, page 269, June 2004.

[MY05] I. Maric and R. D. Yates. Cooperative multicast for maximum network lifetime. *IEEE Journal on Selected Areas in Communications*, 23(1):127–135, Jan. 2005.

[MY10] I. Maric and R. D. Yates. Bandwidth and power allocation for cooperative strategies in gaussian relay networks. *IEEE Transactions on Information Theory*, 56(4):1880–1889, April 2010.

[Nee03] M. J. Neely. *Dynamic power allocation and routing for satellite and wireless networks with time varying channels*. PhD thesis, Massachusetts Institute of Technology, 2003.

[Nee06] M. J. Neely. Energy optimal control for time-varying wireless networks. *IEEE Transactions on Information Theory*, 52(7):2915–2934, July 2006.

[Nee09] M. J. Neely. Stochastic optimization for markov modulated networks with application to delay constrained wireless scheduling. In *Proceedings of the 48th IEEE Conference on Decision and Control*, pages 4826–4833, Dec. 2009.

[Nee10a] M. J. Neely. Dynamic optimization and learning for renewal systems. In *Proc. Asilomar Conf. on Signals, Systems, and Computers*, Nov. 2010.

[Nee10b] M. J. Neely. *Stochastic Network Optimization with Application to Communication and Queueing Systems*. Morgan & Claypool, 2010.

[NML08] M. J. Neely, E. Modiano, and C.-P. Li. Fairness and optimal stochastic control for heterogeneous networks. *IEEE/ACM Transactions on Networking*, 16:396–409, April 2008.

[NMR05] M. J. Neely, E. Modiano, and C. E. Rohrs. Dynamic power allocation and routing for time varying wireless networks. *IEEE Journal on Selected Areas in Communications*, 23(1):89–103, Jan. 2005.

[NU07] M. J. Neely and R. Urgaonkar. Cross-layer adaptive control for wireless mesh networks. *Ad Hoc Networks*, 5:719–743, Aug. 2007.

[NU09] M. J. Neely and R. Urgaonkar. Optimal backpressure routing for wireless networks with multi-receiver diversity. *Ad Hoc Networks*, 7:862–881, July 2009.

[Put05] Martin L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc., New York, 2005.

[PZZ06] C. Peng, H. Zheng, and B. Y. Zhao. Utilization and fairness in spectrum assignment for opportunistic spectrum access. *Mobile Networks and Applications*, 11:555–576, August 2006.

[RE10] B. Rong and A. Ephremides. Network-level cooperation with enhancements based on the physical layer. In *Proc. IEEE Information Theory Workshop*, Jan. 2010.

[Ros96] Sheldon M. Ross. *Stochastic Processes*. John Wiley & Sons, Inc., New York, 1996.

[SBNS07] O. Simeone, Y. Bar-Ness, and U. Spagnolini. Stable throughput of cognitive radios with and without relaying capability. *IEEE Transactions on Communications*, 55(12):2351–2360, Dec. 2007.

[SEA03a] A. Sendonaris, E. Erkip, and B. Aazhang. User cooperation diversity-part i: System description. *IEEE Transactions on Communications*, 51(11):1927–1938, Nov. 2003.

[SEA03b] A. Sendonaris, E. Erkip, and B. Aazhang. User cooperation diversity-part ii: Implementation aspects and performance analysis. *IEEE Transactions on Communications*, 51(11):1939–1948, Nov. 2003.

[SGL06] A. Scaglione, D. L. Goeckel, and J. N. Laneman. Cooperative communications in mobile ad hoc networks. *IEEE Signal Processing Magazine*, 23(5):18–29, Sept. 2006.

[SH08] Y. Shi and Y. T. Hou. A distributed optimization algorithm for multi-hop cognitive radio networks. In *Proc. IEEE INFOCOM*, pages 1292–1300, April 2008.

[Sho04] A. Shokrollahi. Raptor codes. In *Proc. International Symposium on Information Theory*, page 36, July 2004.

[SMS07] B. Sirkeci-Mergen and A. Scaglione. On the power efficiency of cooperative broadcast in dense wireless networks. *IEEE Journal on Selected Areas in Communications*, 25(2):497–507, Feb. 2007.

[SMSM06] B. Sirkeci-Mergen, A. Scaglione, and G. Mergen. Asymptotic analysis of multistage cooperative broadcast in wireless networks. *IEEE Transactions on Information Theory*, 52(6):2531–2550, June 2006.

[SR09] K. Sundaresan and S. Rangarajan. Efficient resource management in ofdma femto cells. In *Proceedings of the tenth ACM international symposium on Mobile ad hoc networking and computing*, MobiHoc '09, pages 33–42, New York, NY, USA, 2009. ACM.

[SSH+10] S. Sharma, Y. Shi, Y. T. Hou, H. D. Sherali, and S. Kompella. Cooperative communications in multi-hop wireless networks: Joint flow routing and relay node assignment. In *Proc. IEEE INFOCOM*, pages 1–9, March 2010.

[SSS+08] O. Simeone, I. Stanojev, S. Savazzi, Y. Bar-Ness, U. Spagnolini, and R. Pickholtz. Spectrum leasing to cooperating secondary ad hoc networks. *IEEE Journal on Selected Areas in Communications*, 26(1):203–213, Jan. 2008.

[TE92] L. Tassiulas and A. Ephremides. Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks. *IEEE Transactions on Automatic Control*, 37(12):1936–1948, Dec. 1992.

[TSM09] R. Tandra, A. Sahai, and S. M. Mishra. What is a spectrum hole and what does it take to recognize one? *Proceedings of the IEEE*, 97(5):824–848, May 2009.

[TV05] D. Tse and P. Viswanath. *Fundamentals of Wireless Communication*. Cambridge University Press, 2005.

[WLX05] W. Wang, X. Liu, and H. Xiao. Exploring opportunistic spectrum availability in wireless communication networks. In *Proc. IEEE VTC*, Sept. 2005.

[WSP07] X. Wu, R. Srikant, and J. R. Perkins. Scheduling efficiency of distributed greedy scheduling algorithms in wireless networks. *IEEE Transactions on Mobile Computing*, 6(6):595–605, June 2007.

[XL10] H. Xu and B. Li. Efficient resource allocation with flexible channel cooperation in ofdma cognitive radio networks. In *Proc. IEEE INFOCOM*, pages 1–9, March 2010.

[YB07] E. M. Yeh and R. A. Berry. Throughput optimal control of cooperative relay networks. *IEEE Transactions on Information Theory*, 53(10):3827–3833, Oct. 2007.

[YBC+07] Y. Yuan, P. Bahl, R. Chandra, T. Moscibroda, and Y. Wu. Allocating dynamic time-spectrum blocks in cognitive radio networks. In *Proceedings of the 8th ACM international symposium on Mobile ad hoc networking and computing*, MobiHoc '07, pages 130–139, New York, NY, USA, 2007.

[YMMZ08] R. Yim, N. Mehta, A. Molisch, and J. Zhang. Progressive accumulative routing: Fundamental concepts and protocol. *IEEE Transactions on Wireless Communications*, 7(11):4142–4154, Nov. 2008.

[ZAL07] Y. Zhao, R. Adve, and T. J. Lim. Improving amplify-and-forward relay networks: Optimal power allocation versus selection. *IEEE Transactions on Wireless Communications*, 6(8):3114–3123, Aug. 2007.

[ZS07] Q. Zhao and B. M. Sadler. A survey of dynamic spectrum access. *IEEE Signal Processing Magazine*, 24(3):79–89, May 2007.

[ZTSC07] Q. Zhao, L. Tong, A. Swami, and Y. Chen. Decentralized cognitive mac for opportunistic spectrum access in ad hoc networks: A pomdp framework. *IEEE Journal on Selected Areas in Communications*, 25(3):589–600, April 2007.

[ZV05] B. Zhao and M. C. Valenti. Practical relay networks: A generalization of hybrid-arq. *IEEE Journal on Selected Areas in Communications*, 23(1):7–18, Jan. 2005.

[ZZ09] J. Zhang and Q. Zhang. Stackelberg game for utility-based cooperative cognitive radio networks. In *Proceedings of the tenth ACM international symposium on Mobile ad hoc networking and computing*, MobiHoc '09, pages 23–32, New York, NY, USA, 2009. ACM.

Appendix A

Appendices for Chapter 2

A.1 Lyapunov Drift under policy STAT

Here, we use "delayed" queue backlogs to express the Lyapunov drift of the CNC algorithm in a form that fits (2.19). Recall that $R_n^{STAT}(t)$ and $\mu_{nm}^{STAT}(t)$ denote the resource allocation decisions under the stationary, randomized policy $STAT$ introduced in Sec. 2.5.2. We use the following sample path inequalities. Specifically, for all $t > d$, we have for each secondary user queue $Q_n(t)$ and for each collision queue $X_m(t)$:

$$Q_n(t-d) + dA_{max} \geq Q_n(t) \geq Q_n(t-d) - d$$
$$X_m(t-d) + d \geq X_m(t) \geq X_m(t-d) - d\rho_m$$

These follow by noting that the queue backlog at time t cannot be smaller than the queue backlog at time $(t-d)$ minus the maximum possible departures in duration $(t-d, d)$. Similarly, it cannot be larger than the queue backlog at time $(t-d)$ plus the maximum possible arrivals in duration $(t-d, d)$. Using these in (2.29) and using $\mathbb{E}\left\{R_n^{STAT}(t)\right\} = r_n^*$ (from (2.25)), we get:

$$\begin{aligned}
\Delta^{CNC}(t) - V\mathbb{E}\left\{\sum_{n=1}^{N} \theta_n R_n^{CNC}(t)\right\} &\leq B + C_U + C_X \\
&- \mathbb{E}\left\{\sum_{n=1}^{N} Q_n(t-d)\Big(\sum_{m=1}^{M} \mu_{nm}^{STAT}(t) S_m(t) - R_n^{STAT}(t)\Big)\right\} \\
&- \mathbb{E}\left\{\sum_{m=1}^{M} X_m(t-d)(\rho_m 1_m(t) - \hat{C}_m^{STAT}(t))\right\} - V\sum_{n=1}^{N} \theta_n r_n^* \qquad \text{(A.1)}
\end{aligned}$$

where C_U and C_X are given by:

$$C_U \triangleq dMN + dA_{max}^2 N \qquad \text{(A.2)}$$

$$C_X \triangleq d\sum_{m=1}^{M}(1+\rho_m^2) \qquad \text{(A.3)}$$

Using iterated expectations, we have the following:

$$\mathbb{E}\left\{\sum_{n=1}^{N} Q_n(t-d)\sum_{m=1}^{M}\mu_{nm}^{STAT}(t)S_m(t)\right\} =$$
$$\mathbb{E}\left\{\sum_{n=1}^{N} Q_n(t-d)\times\mathbb{E}\left\{\sum_{m=1}^{M}\mu_{nm}^{STAT}(t)S_m(t)|\mathcal{T}(t-d)\right\}\right\} \tag{A.4}$$
$$\mathbb{E}\left\{\sum_{m=1}^{M} X_m(t-d)(\rho_m 1_m(t)-\hat{C}_m^{STAT}(t))\right\} =$$
$$\mathbb{E}\left\{\sum_{m=1}^{M} X_m(t-d)\times\mathbb{E}\left\{\rho_m 1_m(t)-\hat{C}_m^{STAT}(t)|\mathcal{T}(t-d)\right\}\right\} \tag{A.5}$$

where $\mathcal{T}(t-d) = (\boldsymbol{H}(t-d), \chi(t-d), \boldsymbol{Q}(t-d))$ represents the composite system state at time $(t-d)$ and includes the topology state and queue backlogs.

By the Markovian property of the $\boldsymbol{H}(t), \chi(t)$ (and therefore $\boldsymbol{P}(t)$) processes, any functionals of these states converge exponentially fast to their steady state values (this is formalized in Appendix A.2). Since the policy $STAT$ makes control decisions only as a function of $\boldsymbol{P}(t)$ and $\boldsymbol{H}(t)$, the resulting allocations are functionals of these Markovian processes. Thus, there exist positive constants α_1, α_2 and $0 < \gamma_1, \gamma_2 < 1$ such that:

$$\mathbb{E}\left\{\sum_{m=1}^{M}\mu_{nm}^{STAT}(t)S_m(t)|\mathcal{T}(t-d)\right\} \geq \mu_n^{STAT} - \alpha_1\gamma_1^d$$
$$\mathbb{E}\left\{\rho_m 1_m(t)-\hat{C}_m^{STAT}(t)|\mathcal{T}(t-d)\right\} \leq \rho_m\nu_m - \hat{c}_m^{STAT} + \alpha_2\gamma_2^d$$

where $\mu_n^{STAT}, \hat{c}_m^{STAT}$ are the steady state values as defined in $(2.26), (2.27)$. Using these, the above can be written as:

$$\mathbb{E}\left\{\sum_{m=1}^{M}\mu_{nm}^{STAT}(t)S_m(t)|\mathcal{T}(t-d)\right\} \geq r_n^* - \alpha_1\gamma_1^d \tag{A.6}$$
$$\mathbb{E}\left\{\rho_m 1_m(t)-\hat{C}_m^{STAT}(t)|\mathcal{T}(t-d)\right\} \leq \alpha_2\gamma_2^d \tag{A.7}$$

Thus, using $(A.6), (A.7)$ in $(A.4), (A.5)$, inequality $(A.1)$ can be expressed as:

$$\Delta^{CNC}(t) - V\mathbb{E}\left\{\sum_{n=1}^{N}\theta_n R_n^{CNC}(t)\right\}$$
$$\leq B + C_U + C_X + \mathbb{E}\left\{\sum_{n=1}^{N} Q_n(t-d)\alpha_1\gamma_1^d\right\} + \mathbb{E}\left\{\sum_{m=1}^{M} X_m(t-d)\alpha_2\gamma_2^d\right\} - V\sum_{n=1}^{N}\theta_n r_n^*$$
$$\leq B + C_U + C_X + NQ_{max}\alpha_1\gamma_1^d + MX_{max}\alpha_2\gamma_2^d - V\sum_{n=1}^{N}\theta_n r_n^*$$

The last step follows from the bounds on $Q_n(t-d)$ and $X_m(t-d)$ established in (2.15) and (2.17).

Define $d_1 = \frac{\log(\alpha_1 U_{max})}{\log(1/\gamma_1)}$, $d_2 = \frac{\log(\alpha_2 X_{max})}{\log(1/\gamma_2)}$. Then choosing $d = \max(d_1, d_2)$, we have:

$$\Delta^{CNC}(t) - V\mathbb{E}\left\{\sum_{n=1}^{N}\theta_n R_n^{CNC}(t)\right\} \leq B + C_U + C_X + N + M - V\sum_{n=1}^{N}\theta_n r_n^* \tag{A.8}$$

Since Q_{max} and X_{max} are $O(V)$, we have $d \sim O(\log V)$.

A.2 Convergence of Markov Chains

Let $Z(t)$ be a finite state, discrete time ergodic Markov Chain. Let $\mathcal{S}$ denote its state space and let $\{\pi_i\}_{i\in\mathcal{S}}$ be the steady state probability distribution. Then, for all integers $d \geq 0$, there exist constants α, γ such that:

$$|Pr\{Z(t) = j|Z(t-d) = i\} - \pi_j| \leq \alpha\gamma^d \tag{A.9}$$

where $\alpha \geq 0$ and $0 < \gamma < 1$. This implies that the Markov Chain converges to its steady state probability distribution exponentially fast (see [Ros96]).

Let $f(Z(t))$ be a positive random function of $Z(t)$ (negative case can be treated similarly). Define $\overline{f} = \sum_{j\in\mathcal{S}}\pi_j m_j$ where $m_j \triangleq \mathbb{E}\{f(Z(t))|Z(t) = j\}$. Then:

$$\begin{aligned}
\mathbb{E}\{f(Z(t))|Z(t-d) = i\} &= \sum_{j\in\mathcal{S}}\mathbb{E}\{f(Z(t))|Z(t) = j\}\, Pr\{Z(t) = j|Z(t-d) = i\} \\
&\leq \sum_{j\in\mathcal{S}} m_j(\pi_j + \alpha\gamma^d) \qquad \text{(using (A.9))} \\
&\leq \overline{f} + s m_{max}\alpha\gamma^d
\end{aligned}$$

where $m_{max} \triangleq \max_{j\in\mathcal{S}} m_j$ and $s = \text{card}\{\mathcal{S}\}$. This shows that functionals of the states of a finite state ergodic Markov Chain converge to their steady state value exponentially fast.

A.3 On Greedy Maximal Weight Matchings

Here, we prove property (2.31) for Greedy Maximal Weight Matchings (GMM) on a weighted graph. While we need this property to hold only for bipartite graphs, it is true in general for arbitrary graphs with non-negative weights.

Let $G = (V, E)$ be a graph with vertices V and edges E. Let w_e denote the weight of an edge $e \in E$. We assume that $w_e \geq 0\ \forall e \in E$. Let $C^{MWM}(G)$ denote the value of the Maximum Weight Match on G and let n be its size. Also, let $C^{GMM}(G)$ denote the value of a Greedy Maximal Weight Match on G. Note that the size of any Greedy Maximal Weight Match must be at least $n/2$. This is true because GMMs have the maximal property, and any maximal match has a size that is at least a factor of 2 away from the size of any other maximal match. We have the following:

Claim: $C^{MWM}(G) \leq 2C^{GMM}(G)$

Proof: Suppose w_1 is the weight of the first edge e_1 that is chosen by the greedy procedure (as described in Sec. 2.6) while constructing a Greedy Maximal Weight Match on G. Then we know that w_1 is also the maximum edge weight in G. Once e_1 is chosen, all edges that share a common vertex with it are labeled "inactive" and are not considered for addition into the match. This means that at most 2 edges of the Maximum Weight Match may be labeled inactive. Further, the sum of their weights cannot exceed $2w_1$. The other $(n-2)$ or more edges of the Maximum Weight Match are candidates for selection during the next iteration of the greedy procedure. This argument can be repeated for each of the first $n/2$ iterations of the greedy procedure and yields

$$C^{MWM}(G) \leq 2 \sum_{i=1}^{n/2} w_i \leq 2C^{GMM}(G)$$

Appendix B

Appendices for Chapter 3

B.1 Proof of Theorem 4

Here, we prove Theorem 4 by comparing the Lyapunov drift of the dynamic control algorithm (3.7) with that of an optimal stationary, randomized policy. Let r_s^* and e_i^* $\forall i \in \widehat{\mathcal{R}}$ denote the optimal value of the objective in (3.2). Then the following fact can be shown using the techniques developed in [Nee06]

Existence of an Optimal Stationary, Randomized Policy: Assuming i.i.d. $\mathcal{T}(t)$ states, there exists a stationary randomized policy π that chooses feasible control action $\mathcal{I}^\pi(t)$ and power allocations $P_i^\pi(t)$ for all $i \in \widehat{\mathcal{R}}$ every slot purely as a function of the current channel state $\mathcal{T}(t)$ and yields the following for some $\epsilon > 0$:

$$\mathbb{E}\left\{\Phi_s^\pi(t)\right\} \geq \rho_s\lambda_s + \epsilon \tag{B.1}$$

$$\mathbb{E}\left\{P_i^\pi(t)\right\} + \epsilon \leq P_i^{avg} \tag{B.2}$$

$$\alpha_s\mathbb{E}\left\{\Phi_s^\pi(t)\right\} - \sum_{i\in\mathcal{N}}\beta_i\mathbb{E}\left\{P_i^\pi(t)\right\} = \alpha_s r_s^* - \sum_{i\in\mathcal{N}}\beta_i e_i^* \tag{B.3}$$

Let $\boldsymbol{Q}(t) = (Z_s(t), X_i(t))$ $\forall i \in \widehat{\mathcal{R}}$ represent the collection of these queue backlogs in timeslot t. We define a quadratic Lyapunov function:

$$L(\boldsymbol{Q}(t)) \triangleq \frac{1}{2}\left[Z_s^2(t) + \sum_{i\in\widehat{\mathcal{R}}} X_i^2(t)\right]$$

Also define the conditional Lyapunov drift $\Delta(\boldsymbol{Q}(t))$ as follows:

$$\Delta(\boldsymbol{Q}(t)) \triangleq \mathbb{E}\left\{L(\boldsymbol{Q}(t+1)) - L(\boldsymbol{Q}(t)) | \boldsymbol{Q}(t)\right\}$$

Using queueing dynamics (3.5), (3.6), the Lyapunov drift under any control policy can be computed as follows:

$$\Delta(\boldsymbol{Q}(t)) \leq B - Z_s(t)\mathbb{E}\left\{\Phi_s(t) - \rho_s A_s(t) | \boldsymbol{Q}(t)\right\} - \sum_{i\in\widehat{\mathcal{R}}} X_i(t)\mathbb{E}\left\{P_i^{avg} - P_i(t) | \boldsymbol{Q}(t)\right\} \tag{B.4}$$

where $B = \frac{1+\lambda_s^2\rho_s^2 + \sum_{i\in\widehat{\mathcal{R}}}(P_i^{avg})^2 + (P^{max})^2}{2}$.

For a given control parameter $V \geq 0$, from both sides of the above inequality we subtract a "reward" metric $V\mathbb{E}\left\{\alpha_s \Phi_s(t) - \sum_{i \in \widehat{\mathcal{R}}} \beta_i P_i(t) | \boldsymbol{Q}(t)\right\}$ to get the following:

$$\Delta(\boldsymbol{Q}(t)) - V\mathbb{E}\left\{\alpha_s \Phi_s(t) - \sum_{i \in \widehat{\mathcal{R}}} \beta_i P_i(t) | \boldsymbol{Q}(t)\right\} \leq B - Z_s(t)\mathbb{E}\left\{\Phi_s(t) - \rho_s A_s(t) | \boldsymbol{Q}(t)\right\}$$
$$- \sum_{i \in \widehat{\mathcal{R}}} X_i(t) \mathbb{E}\left\{P_i^{avg} - P_i(t) | \boldsymbol{Q}(t)\right\} - V\mathbb{E}\left\{\alpha_s \Phi_s(t) - \sum_{i \in \widehat{\mathcal{R}}} \beta_i P_i(t) | \boldsymbol{Q}(t)\right\} \tag{B.5}$$

From the above, it can be seen that the dynamic control algorithm (3.7) is designed to take a control action that minimizes the right hand side of (B.5) over all possible options every slot, including the stationary policy π. Thus, using (B.1), (B.2), (B.3), we can write the above as:

$$\Delta(\boldsymbol{Q}(t)) - V\mathbb{E}\left\{\alpha_s \Phi_s(t) - \sum_{i \in \widehat{\mathcal{R}}} \beta_i P_i(t) | \boldsymbol{Q}(t)\right\} \leq B - Z_s(t)\epsilon - \sum_{i \in \widehat{\mathcal{R}}} X_i(t)\epsilon - V\alpha_s r_s^* - \sum_{i \in \widehat{\mathcal{R}}} \beta_i e_i^* \tag{B.6}$$

Theorem 1 now follows by a direct application of the Lyapunov optimization Theorem [GNT06].

B.2 Solution to (3.17) using KKT conditions

We ignore the constant terms in the objective. It is easy to see that the first constraint in (3.17) must be met with equality. The Lagrangian is given by:

$$\begin{aligned}\mathcal{L} = &(X_s + V\beta_s)P_s + \sum_{i \in \mathcal{U}_k} (X_i + V\beta_i)P_i - \lambda_s(P_s - P_s^{\mathcal{U}_k}) \\ &- \sum_{i \in \mathcal{U}_k} \lambda_i P_i + \beta_s(P_s - P_s^{max}) + \sum_{i \in \mathcal{U}_k} \beta_i (P_i - P_i^{max}) \\ &+ \nu\Big[\log(1 + \theta_s P_s) + \sum_{i \in \mathcal{U}_k} \log(1 + \theta_i P_i) - \frac{mR}{W}\Big]\end{aligned}$$

where $\theta_s = \frac{m}{W}|h_{sd}|^2, \theta_i = \frac{m}{W}|h_{id}|^2$. The KKT conditions for all $i \in \mathcal{U}_k$ are [BV04]:

$$\begin{aligned}&\lambda_s^*(P_s^* - P_s^{\mathcal{U}_k}) = 0 \qquad \lambda_i^* P_i^* = 0 \\ &\beta_s^*(P_s^* - P_s^{max}) = 0 \qquad \beta_i^*(P_i^* - P_i^{max}) = 0 \\ &\lambda_s^*, \lambda_i^*, \beta_s^*, \beta_i^* \geq 0 \\ &(X_s + V\beta_s) - \lambda_s^* + \beta_s^* + \frac{\nu^* \theta_s}{1 + \theta_s P_s^*} = 0 \\ &(X_i + V\beta_i) - \lambda_i^* + \beta_i^* + \frac{\nu^* \theta_i}{1 + \theta_i P_i^*} = 0\end{aligned}$$

If $\nu^* > 0$, then we must have that $\lambda_s^* - \beta_s^* > 0$ and $\lambda_i^* - \beta_i^* > 0$ for all i. This would mean that $P_s^* = P_s^{\mathcal{U}_k}$ and $P_i^* = 0$. For some $\nu^* \leq 0$, we have three cases:

1. If $\lambda_i^* = \beta_i^*$, we get $P_i^* = \frac{-\nu^*}{X_i + V\beta_i} - \frac{1}{\theta_i}$
2. If $\lambda_i^* > \beta_i^*$, then we must have $\lambda_i^* > 0$ and we get $P_i^* = 0$
3. If $\lambda_i^* < \beta_i^*$, then we must have $\beta_i^* > 0$ and we get $P_i^* = P_i^{max}$

Similar results can be obtained for P_s^*. Combining these, we get:

$$P_s^* = \Big[\frac{-\nu*}{X_s + V\beta_s} - \frac{1}{\theta_s}\Big]_{P_s^{\mathcal{U}_k}}^{P_s^{max}}, \qquad P_i^* = \Big[\frac{-\nu*}{X_i + V\beta_i} - \frac{1}{\theta_i}\Big]_0^{P_i^{max}}$$

where $[X]_0^{P_{max}}$ denotes $\min[\max(X, 0), P_{max}]$.

B.3 Solution to (3.21) using KKT conditions

It is easy to see that the first constraint in (3.21) must be met with equality. The Lagrangian is given by:

$$\begin{aligned}\mathcal{L} = &\sum_{i\in\mathcal{R}_s}(X_i + V\beta_i)P_i - \sum_{i\in\mathcal{R}_s}\lambda_i P_i + \sum_{\in\mathcal{R}_s}\beta_i(P_i - P_i^{max}) \\ &+ \nu\Big[\sum_{\in\mathcal{R}_s}\frac{P_s^2|h_{si}|^4 + P_s|h_{si}|^2 W/m}{|h_{si}|^2P_s + |h_{id}|^2P_i + W/m} - \theta'\Big]\end{aligned}$$

The KKT conditions for all $i \in \mathcal{R}_s$ are:

$$\begin{aligned}&\lambda_i^* P_i^* = 0 \qquad \beta_i^*(P_i^* - P_i^{max}) = 0 \qquad \lambda_i^*, \beta_i^* \geq 0 \\ &(X_i + V\beta_i) - \lambda_i^* + \beta_i^* = \frac{\nu^*|h_{id}|^2(P_s^2|h_{si}|^4 + P_s|h_{si}|^2W/m)}{(|h_{si}|^2P_s + |h_{id}|^2P_i^* + W/m)^2}\end{aligned}$$

If $\nu^* < 0$, then we must have that $\lambda_i^* - \beta_i^* > 0$ for all i. This would mean that $P_i^* = 0$. For some $\nu^* \geq 0$, we have three cases:

1. If $\lambda_i^* = \beta_i^*$, we get $P_i^* = \sqrt{\frac{\nu^*(P_s^2|h_{si}|^4 + P_s|h_{si}|^2W/m)}{(X_i + V\beta_i)|h_{id}|^2}} - \frac{P_s|h_{si}|^2 + W/m}{|h_{id}|^2}$
2. If $\lambda_i^* > \beta_i^*$, then we must have $\lambda_i^* > 0$ and we get $P_i^* = 0$
3. If $\lambda_i^* < \beta_i^*$, then we must have $\beta_i^* > 0$ and we get $P_i^* = P_i^{max}$

Combining these, we get:

$$P_i^* = \Big[\sqrt{\frac{\nu^*(P_s^2|h_{si}|^4 + P_s|h_{si}|^2W/m)}{(X_i + V\beta_i)|h_{id}|^2}} - \frac{P_s|h_{si}|^2 + W/m}{|h_{id}|^2}\Big]_0^{P_i^{max}}$$

where $[X]_0^{P_{max}}$ denotes $\min[\max(X, 0), P_{max}]$.

Appendix C

Appendices for Chapter 4

C.1 Proof of Lemma 3

Let $Q_{su}^{fab}(t)$ denote the queue backlog value under the *Frame-Based-Drift-Plus-Penalty-Algorithm* for all $t \in \{t_k, t_k + 1, \ldots, t_{k+1} - 1\}$. Then, since the admission control decision (4.15) of the *Frame-Based-Drift-Plus-Penalty-Algorithm* minimizes the term $(Q_{su}(t) - V)R_{su}(t)$ for all $Q_{su}(t)$, we have:

$$\mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1} (Q_{su}^{fab}(t) - V)R_{su}^{alt}(t)|\boldsymbol{Q}(t_k)\right\} \geq \mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1} (Q_{su}^{fab}(t) - V)R_{su}^{fab}(t)|\boldsymbol{Q}(t_k)\right\} \quad \text{(C.1)}$$

Note that we are not implementing the admission control decisions of *ALT* in the left hand side of the above.

Next, we make use of the following sample path relations in (C.1) to prove (4.39). For all $t \in \{t_k, t_k + 1, \ldots, t_{k+1} - 1\}$, the following hold under any control algorithm:

$$Q_{su}(t_k) \geq Q_{su}(t) - (t - t_k)A_{max} \quad \text{(C.2)}$$

$$Q_{su}(t_k) \leq Q_{su}(t) + (t - t_k)\mu_{max} \quad \text{(C.3)}$$

(C.2) follows by noting that the maximum number of arrivals to the secondary user queue in the interval $[t_k, \ldots, t)$ is at most $(t - t_k)A_{max}$. Similarly, (C.3) follows by noting that the maximum number of departures from the secondary user queue in the interval $[t_k, \ldots, t)$ is at most $(t - t_k)\mu_{max}$.

Using (C.2) in the left hand side of (C.1) yields:

$$\mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1} (Q_{su}^{fab}(t) - V)R_{su}^{alt}(t)|\boldsymbol{Q}(t_k)\right\} \leq$$

$$\mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1} (Q_{su}(t_k) - V)R_{su}^{alt}(t)|\boldsymbol{Q}(t_k)\right\} + \mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1} (t - t_k)A_{max}R_{su}^{alt}(t)|\boldsymbol{Q}(t_k)\right\}$$

Using the fact that $R_{su}^{alt}(t) \leq A_{max}$ and $\sum_{t=t_k}^{t_{k+1}-1}(t-t_k) = \frac{T[k](T[k]-1)}{2}$, we get:

$$\mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1}(Q_{su}^{fab}(t)-V)R_{su}^{alt}(t)|\boldsymbol{Q}(t_k)\right\} \leq$$
$$\mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1}(Q_{su}(t_k)-V)R_{su}^{alt}(t)|\boldsymbol{Q}(t_k)\right\} + \frac{DA_{max}^2}{2} \tag{C.4}$$

Next, using (C.3) in the right hand side of (C.1) yields:

$$\mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1}(Q_{su}^{fab}(t)-V)R_{su}^{fab}(t)|\boldsymbol{Q}(t_k)\right\} \geq$$
$$\mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1}(Q_{su}(t_k)-V)R_{su}^{fab}(t)|\boldsymbol{Q}(t_k)\right\} - \mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1}(t-t_k)\mu_{max}R_{su}^{fab}(t)|\boldsymbol{Q}(t_k)\right\}$$

Again using the fact that $R_{su}^{fab}(t) \leq A_{max}$ and $\sum_{t=t_k}^{t_{k+1}-1}(t-t[k]) = \frac{T[k](T[k]-1)}{2}$, we get:

$$\mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1}(Q_{su}^{fab}(t)-V)R_{su}^{fab}(t)|\boldsymbol{Q}(t_k)\right\} \geq$$
$$\mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1}(Q_{su}(t_k)-V)R_{su}^{fab}(t)|\boldsymbol{Q}(t_k)\right\} - \frac{D\mu_{max}A_{max}}{2} \tag{C.5}$$

Using (C.4) and (C.5) in (C.1), we have:

$$\mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1}(Q_{su}(t_k)-V)R_{su}^{alt}(t)|\boldsymbol{Q}(t_k)\right\} \geq \mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1}(Q_{su}(t_k)-V)R_{su}^{fab}(t)|\boldsymbol{Q}(t_k)\right\} - C$$

C.2 Proof of Theorem 5, parts (2) and (3)

We prove parts (2) and (3) of Theorem 5 using the technique of Lyapunov optimization. Using (4.14), a bound on the Lyapunov drift under the *Frame-Based-Drift-Plus-Penalty-Algorithm* is given by:

$$\Delta(t_k) - V\mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1} R_{su}^{fab}(t)|\boldsymbol{Q}(t_k)\right\} \leq B + (Q_{su}(t_k) - V)\mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1} R_{su}^{fab}(t)|\boldsymbol{Q}(t_k)\right\}$$
$$- X_{su}(t_k)\mathbb{E}\left\{T[k]P_{avg}|\boldsymbol{Q}(t_k)\right\} - \mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1}(Q_{su}(t_k)\mu_{su}^{fab}(t) - X_{su}(t_k)P_{su}^{fab}(t))|\boldsymbol{Q}(t_k)\right\} \tag{C.6}$$

Using Lemma 3, we have that:

$$\mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1}(Q_{su}(t_k) - V)R_{su}^{fab}(t)|\boldsymbol{Q}(t_k)\right\} \leq C + \mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1}(Q_{su}(t_k) - V)R_{su}^{alt}(t)|\boldsymbol{Q}(t_k)\right\}$$

Next, note that under the ALT algorithm, we have:

$$\frac{\mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1}(Q_{su}(t_k) - V)R_{su}^{alt}(t)|\boldsymbol{Q}(t_k)\right\}}{\mathbb{E}\left\{T[k]|\boldsymbol{Q}(t_k)\right\}} \leq \frac{\mathbb{E}\left\{\sum_{t=t_k}^{\hat{t}_{k+1}-1}(Q_{su}(t_k) - V)R_{su}^{stat}(t)|\boldsymbol{Q}(t_k)\right\}}{\mathbb{E}\left\{\hat{T}[k]|\boldsymbol{Q}(t_k)\right\}}$$

To see this, we have two cases:

1. $Q_{su}(t_k) > V$: Then, $R_{su}^{alt}(t) = 0$ for all $t \in \{t_k, t_k + 1, \ldots, t_{k+1} - 1\}$, so that the left hand side above is 0 while the right hand side is ≥ 0. Hence, the inequality follows.

2. $Q_{su}(t_k) \leq V$: Then, $R_{su}^{alt}(t) = A_{su}(t)$ for all $t \in \{t_k, t_k + 1, \ldots, t_{k+1} - 1\}$, so that the left hand side becomes $(Q_{su}(t_k) - V)\lambda_{su}$ while the right hand side cannot be smaller than $(Q_{su}(t_k) - V)\lambda_{su}$.

Combining these, we get:

$$(Q_{su}(t_k) - V)\mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1} R_{su}^{fab}(t)|\boldsymbol{Q}(t_k)\right\} \leq$$
$$C + (Q_{su}(t_k) - V)\mathbb{E}\left\{\sum_{t=t_k}^{\hat{t}_{k+1}-1} R_{su}^{stat}(t)|\boldsymbol{Q}(t_k)\right\}\frac{\mathbb{E}\left\{T[k]|\boldsymbol{Q}(t_k)\right\}}{\mathbb{E}\left\{\hat{T}[k]|\boldsymbol{Q}(t_k)\right\}}$$

Finally, since the resource allocation part of the *Frame-Based-Drift-Plus-Penalty-Algorithm* maximizes the ratio in (4.16), we have:

$$\mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1}(Q_{su}(t_k)\mu_{su}^{fab}(t) - X_{su}(t_k)P_{su}^{fab}(t))|\boldsymbol{Q}(t_k)\right\} \geq$$
$$\mathbb{E}\left\{\sum_{t=t_k}^{\hat{t}_{k+1}-1}(Q_{su}(t_k)\mu_{su}^{stat}(t) - X_{su}(t_k)P_{su}^{stat}(t))|\boldsymbol{Q}(t_k)\right\}\frac{\mathbb{E}\left\{T[k]|\boldsymbol{Q}(t_k)\right\}}{\mathbb{E}\left\{\hat{T}[k]|\boldsymbol{Q}(t_k)\right\}}$$

Using these in (C.6), we have:

$$\begin{aligned}
&\Delta(t_k) - V\mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1}R_{su}^{fab}(t)|\boldsymbol{Q}(t_k)\right\} \leq B + C \\
&+ (Q_{su}(t_k) - V)\mathbb{E}\left\{\sum_{t=t_k}^{\hat{t}_{k+1}-1}R_{su}^{stat}(t)|\boldsymbol{Q}(t_k)\right\}\frac{\mathbb{E}\left\{T[k]|\boldsymbol{Q}(t_k)\right\}}{\mathbb{E}\left\{\hat{T}[k]|\boldsymbol{Q}(t_k)\right\}} \\
&- \mathbb{E}\left\{\sum_{t=t_k}^{\hat{t}_{k+1}-1}(Q_{su}(t_k)\mu_{su}^{stat}(t) - X_{su}(t_k)P_{su}^{stat}(t))|\boldsymbol{Q}(t_k)\right\}\frac{\mathbb{E}\left\{T[k]|\boldsymbol{Q}(t_k)\right\}}{\mathbb{E}\left\{\hat{T}[k]|\boldsymbol{Q}(t_k)\right\}} \\
&- X_{su}(t_k)\mathbb{E}\left\{T[k]P_{avg}|\boldsymbol{Q}(t_k)\right\}
\end{aligned}$$

Using (4.34)-(4.36) in the inequality above, we get:

$$\Delta(t_k) - V\mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1}R_{su}^{fab}(t)|\boldsymbol{Q}(t_k)\right\} \leq B + C - V\upsilon^*\mathbb{E}\left\{T[k]|\boldsymbol{Q}(t_k)\right\} \tag{C.7}$$

To prove (4.41), we rearrange (C.7) to get:

$$\begin{aligned}
\Delta(t_k) &\leq B + C - V\upsilon^*\mathbb{E}\left\{T[k]|\boldsymbol{Q}(t_k)\right\} + V\mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1}R_{su}^{fab}(t)|\boldsymbol{Q}(t_k)\right\} \\
&\leq B + C + VT_{max}A_{max}
\end{aligned}$$

(4.41) now follows from Theorem 4.1 of [Nee10b]. Since $X_{su}(t_k)$ is mean rate stable, (4.42) follows from Theorem 2.5(b) of [Nee10b].

To prove (4.44), we take expectations of both sides of (C.7) to get:

$$\mathbb{E}\left\{L(\boldsymbol{Q}(t_{k+1}))\right\} - \mathbb{E}\left\{L(\boldsymbol{Q}(t_k))\right\} - V\mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1}R_{su}^{fab}(t)\right\} \leq B + C - V\upsilon^*\mathbb{E}\left\{T[k]\right\}$$

Summing over $k \in \{1, 2, \ldots, K\}$, dividing by V, and rearranging yields:

$$\sum_{k=1}^{K} \mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1} R_{su}^{fab}(t)\right\} \geq \upsilon^* \sum_{k=1}^{K} \mathbb{E}\left\{T[k]\right\} - \frac{(B+C)K}{V}$$

where we used that fact that $\mathbb{E}\left\{L(\boldsymbol{Q}(t_{K+1}))\right\} \geq 0$ and $\mathbb{E}\left\{L(\boldsymbol{Q}(t_1))\right\} = 0$. From this, we have:

$$\frac{\sum_{k=1}^{K} \mathbb{E}\left\{\sum_{t=t_k}^{t_{k+1}-1} R_{su}^{fab}(t)\right\}}{\sum_{k=1}^{K} \mathbb{E}\left\{T[k]\right\}} \geq \upsilon^* - \frac{(B+C)K}{V \sum_{k=1}^{K} \mathbb{E}\left\{T[k]\right\}} \geq \upsilon^* - \frac{B+C}{VT_{min}}$$

since $\sum_{k=1}^{K} \mathbb{E}\left\{T[k]\right\} \geq KT_{min}$. This proves (4.44).

C.3 Computing D

Here, we compute a finite D that satisfies (4.2). First, note that $\mathbb{E}\left\{T^2[k]\right\}$ would be maximum when the secondary user never cooperates. Next, let $I[k]$ and $B[k]$ denote the lengths of the primary user idle and busy periods, respectively, in the k^{th} frame. Thus, we have $T[k] = I[k] + B[k]$.

In the following, we drop $[k]$ from the notation for convenience. Using the independence of I and B, we have:

$$\mathbb{E}\left\{T^2\right\} = \mathbb{E}\left\{I^2\right\} + \mathbb{E}\left\{B^2\right\} + 2\mathbb{E}\left\{I\right\}\mathbb{E}\left\{B\right\}$$

We note that I is a geometric r.v. with parameter λ_{pu}. Thus, $\mathbb{E}\left\{I\right\} = 1/\lambda_{pu}$ and $\mathbb{E}\left\{I^2\right\} = (2 - \lambda_{pu})/\lambda_{pu}^2$. To calculate $\mathbb{E}\left\{B\right\}$, we apply Little's Theorem to get:

$$\mathbb{E}\left\{I\right\} = \Big(1 - \frac{\lambda_{pu}}{\phi_{nc}}\Big)(\mathbb{E}\left\{I\right\} + \mathbb{E}\left\{B\right\})$$

This yields $\mathbb{E}\left\{B\right\} = 1/(\phi_{nc} - \lambda_{pu})$. To calculate $\mathbb{E}\left\{B^2\right\}$, we use the observation that changing the service order of packets in the primary queue to preemptive LIFO does not change the length of the busy period B. However, with LIFO scheduling, B now equals the duration that the first packet stays in the queue. Next, suppose there are N packets that interrupt the service of the first packet. Let these be indexed as $\{1, 2, \ldots, N\}$. We can relate B to the service time X of the first packet and the durations for which all these other packets stay in the queue as follows:

$$B = X + \sum_{i=1}^{N} B_i \tag{C.8}$$

Here, B_i denotes the duration for which packet i stays in the queue. Using the memoryless property of the i.i.d. arrival process of the primary packets as well as the i.i.d. nature of the service times, it follows that all the r.v.'s B_i are i.i.d. with the *same* distribution as B. Further, they are independent of N. Squaring (C.8) and taking expectations, we get:

$$\mathbb{E}\left\{B^2\right\} = \mathbb{E}\left\{X^2\right\} + 2\mathbb{E}\left\{X\right\}\mathbb{E}\left\{N\right\}\mathbb{E}\left\{B\right\} + \mathbb{E}\left\{\Big(\sum_{i=1}^{N} B_i\Big)^2\right\} \tag{C.9}$$

Note that X is a geometric r.v. with parameter ϕ_{nc}. Thus $\mathbb{E}\left\{X\right\} = 1/\phi_{nc}$ and $\mathbb{E}\left\{X^2\right\} = (2-\phi_{nc})/\phi_{nc}^2$. Also, $\mathbb{E}\left\{N\right\} = \lambda_{pu}\mathbb{E}\left\{X\right\} = \lambda_{pu}/\phi_{nc}$. Using these in (C.9), we have:

$$\mathbb{E}\left\{B^2\right\} = \frac{(2-\phi_{nc})}{\phi_{nc}^2} + \frac{2\lambda_{pu}}{\phi_{nc}^2(\phi_{nc}-\lambda_{pu})} + \mathbb{E}\left\{\Big(\sum_{i=1}^{N} B_i\Big)^2\right\} \tag{C.10}$$

To calculate the last term, we have:

$$\begin{aligned}
\mathbb{E}\left\{\Big(\sum_{i=1}^{N} B_i\Big)^2\right\} &= \mathbb{E}\left\{\sum_{i=1}^{N} B_i^2\right\} + 2\mathbb{E}\left\{\sum_{i\neq j} B_iB_j\right\} \\
&= \mathbb{E}\left\{N\right\}\mathbb{E}\left\{B^2\right\} + 2(\mathbb{E}\left\{B\right\})^2(\mathbb{E}\left\{N^2\right\} - \mathbb{E}\left\{N\right\})
\end{aligned}$$

Note that given $X = x$, N is a binomial r.v. with parameters (x, λ_{pu}). Thus, we have:

$$\begin{aligned}
\mathbb{E}\left\{N^2\right\} &= \sum_{x\geq 1}\mathbb{E}\left\{N^2|X=x\right\}\mathrm{Prob}[X=x] = \sum_{x\geq 1}\Big[(x\lambda_{pu})^2 + x\lambda_{pu}(1-\lambda_{pu})\Big](1-\phi_{nc})^{x-1}\phi_{nc} \\
&= \lambda_{pu}^2\sum_{x\geq 1} x^2\phi_{nc}(1-\phi_{nc})^{x-1} + \lambda_{pu}(1-\lambda_{pu})\sum_{x\geq 1} x\phi_{nc}(1-\phi_{nc})^{x-1} \\
&= \lambda_{pu}^2\frac{(2-\phi_{nc})}{\phi_{nc}^2} + \lambda_{pu}(1-\lambda_{pu})\frac{1}{\phi_{nc}}
\end{aligned}$$

Using this, we have:

$$\begin{aligned}
\mathbb{E}\left\{\Big(\sum_{i=1}^{N} B_i\Big)^2\right\} &= \frac{\lambda_{pu}}{\phi_{nc}}\mathbb{E}\left\{B^2\right\} + 2\Big(\frac{1}{\phi_{nc}-\lambda_{pu}}\Big)^2(\mathbb{E}\left\{N^2\right\} - \mathbb{E}\left\{N\right\}) \\
&= \frac{\lambda_{pu}}{\phi_{nc}}\mathbb{E}\left\{B^2\right\} + 2\Big(\frac{1}{\phi_{nc}-\lambda_{pu}}\Big)^2\Big(\frac{2\lambda_{pu}^2(1-\phi_{nc})}{\phi_{nc}^2}\Big)
\end{aligned}$$

Using this in (C.10), we have:

$$\mathbb{E}\left\{B^2\right\} = \frac{(2-\phi_{nc})}{\phi_{nc}^2} + \frac{2\lambda_{pu}}{\phi_{nc}^2(\phi_{nc}-\lambda_{pu})} + \frac{\lambda_{pu}}{\phi_{nc}}\mathbb{E}\left\{B^2\right\} + 2\Big(\frac{1}{\phi_{nc}-\lambda_{pu}}\Big)^2\Big(\frac{2\lambda_{pu}^2(1-\phi_{nc})}{\phi_{nc}^2}\Big)$$

Simplifying this yields:

$$\mathbb{E}\left\{B^2\right\} = \frac{(2-\phi_{nc})}{\phi_{nc}(\phi_{nc}-\lambda_{pu})} + \frac{2\lambda_{pu}}{\phi_{nc}(\phi_{nc}-\lambda_{pu})^2} + \frac{4\lambda_{pu}^2(1-\phi_{nc})}{\phi_{nc}(\phi_{nc}-\lambda_{pu})^3} \tag{C.11}$$

Appendix D

Appendices for Chapter 5

D.1 Proof of Lemma 4

We argue by contradiction. Suppose an optimal solution to (5.2) without the constraint $x \geq 0$ is given by $x' \neq x^*$. Then, we have that $c^T x' < c^T x^*$. Further, x' satisfies all the constraints we did not remove, but must violate at least one of the constraints that we removed. Thus, we have that $Ax' = b$ and $x' \not\geq 0$. Now let x'' be a convex combination of x^* and x', i.e., $x'' = \theta x^* + (1-\theta)x'$ where $0 < \theta < 1$. We have that $c^T x'' = \theta c^T x^* + (1-\theta) c^T x'$. Since $c^T x' < \theta c^T x^* + (1-\theta) c^T x' < c^T x^*$, we have that $c^T x' < c^T x'' < c^T x^*$.

Since x^* satisfies the strict inequality constraint $x > 0$ in all entries, there must be a ball about x^* that still satisfies the constraint $x \geq 0$. Further, the line segment joining x^* and x' intersects this ball. Let us choose θ such that x'' is this point of intersection. Then x'' still satisfies the constraint $x'' \geq 0$. However, $c^T x'' < c^T x^*$, which contradicts the fact that x^* solves (5.2) optimally.

D.2 Proof of Lemma 5

Consider the line network as shown in Fig. 5.4. We first show that the optimal cooperating set cannot contain any relay node that lies to the left of the source. Suppose the optimal set contains one or more such nodes. Then, we can replace all transmissions by these nodes with a source transmission and get a smaller delay. This is because the source has a strictly higher transmission capacity to *all* nodes to its right than each of these nodes.

Next, we show that the optimal cooperating set must contain all the nodes that are located between s and d. We know that s is the first node to transmit. The first relay node that decodes the packet is node 1, since link $s-1$ has the smallest distance and therefore the highest transmission capacity among all links from s to nodes to the right of s. From Theorem 6, we know that once node 1 has decoded the packet, it should start transmitting if it is part of the optimal set. Else, it never transmits and the source continues to transmit until another node can decode the packet. Suppose that the optimal set does not contain node 1. Then, we can get a smaller delay by having node 1 transmit instead of s once it has decoded the packet. This is because node 1 has a strictly higher transmission capacity to *all* nodes to its right than s. Thus, we have that the optimal set must contain node 1.

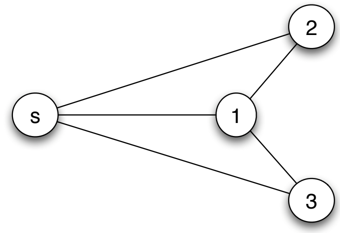

Figure D.1: The 4 node example network used in Appendix D.3.

The above argument can now be applied to each of the nodes $2, 3, \ldots, n$ as in Fig. 5.4. This proves the Lemma.

D.3 A Simple Example

Here, we show an example where different power levels can give rise to different decoding orders for the same relay set under the greedy transmission strategy when the rate-power curve is non-linear. Consider the 4 node network in Fig. D.1. We assume the rate-power curves on all links except link $s-3$ are linear. Specifically, $C_{ij}(P_i) = h_{ij}P_i$ for all $ij \neq s3$. However, the rate-power curve on link $s-3$ is logarithmic and is given by $C_{s3}(P_s) = \log(1 + h_{s3}P_s)$.

Next, suppose $h_{s1} > h_{s2}, h_{s3}$ and $h_{12} = h_{13}$. Also, let $I_{max} = 1$. Then, node 1 is the first node to decode the packet for all $P_s > 0$. Also, we have $\Delta_0 = \frac{1}{C_{s1}(P_s)} = \frac{1}{h_{s1}P_s}$.

The mutual information state at nodes 2 and 3 at the end of stage 0 is given by $I_2(t_1) = \Delta_0 C_{s2}(P_s) = \Delta_0 h_{s2}P_s$ and $I_3(t_1) = \Delta_0 C_{s3}(P_s) = \Delta_0 \log(1 + h_{s3}P_s)$ respectively. Under the greedy transmission strategy, after stage 0, node 1 will continue to transmit until any of nodes 2 or 3 decodes the packet. Suppose node 1 uses transmit power $P_1 > 0$. Then, the time for node 2 to decode if node 1 continues to transmit is given by:

$$\delta_2 = \frac{I_{max} - I_2(t_1)}{C_{12}(P_1)} = \frac{1 - \Delta_0 h_{s2}P_s}{h_{12}P_1} = \frac{1 - \frac{h_{s2}}{h_{s1}}}{h_{12}P_1}$$

Similarly, the time for node 3 to decode if node 1 continues to transmit is given by:

$$\delta_3 = \frac{I_{max} - I_3(t_1)}{C_{13}(P_1)} = \frac{1 - \Delta_0 \log(1 + h_{s3}P_s)}{h_{13}P_1} = \frac{1 - \frac{\log(1+h_{s3}P_s)}{h_{s1}P_s}}{h_{13}P_1}$$

Since $h_{12} = h_{13}$, from the above we have that $\delta_2 > \delta_3$ if $h_{s2}P_s < \log(1 + h_{s3}P_s)$ and $\delta_2 < \delta_3$ if $h_{s2}P_s > \log(1 + h_{s3}P_s)$. Let $h_{s2} = 0.05, h_{s3} = 0.1$. Then, for $P_s = 1$, we get $\delta_2 < \delta_3$ since $0.05 < \log(1.1)$. However, for $P_s = 100$, we have that $\delta_2 < \delta_3$ since $5 > \log(11)$. This shows that different power levels can give rise to different decoding orders for the same relay set under the greedy transmission strategy when the rate-power curve is non-linear.