Optimal Power Allocation and Scheduling of Real-Time Data for Cognitive Radios

by

Ahmed Emad Ewaisha

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

Approved October 2016 by the
Graduate Supervisory Committee:

Cihan Tepedelenlioğlu, Chair
Lei Ying
Daniel Bliss
Oliver Kosut

ARIZONA STATE UNIVERSITY

December 2016

ABSTRACT

In this dissertation, I propose potential techniques to improve the quality-of-service (QoS) of real-time applications in cognitive radio (CR) systems. Unlike best-effort applications, real-time applications, such as audio and video, have a QoS that need to be met. There are two different frameworks that are used to study the QoS in the literature, namely, the average-delay and the hard-deadline frameworks. In the former, the scheduling algorithm has to guarantee that the packet's average delay is below a prespecified threshold while the latter imposes a hard deadline on each packet in the system. In this dissertation, I present joint power allocation and scheduling algorithms for each framework and show their applications in CR systems which are known to have strict power limitations so as to protect the licensed users from interference.

A common aspect of the two frameworks is the packet service time. Thus, the effect of multiple channels on the service time is studied first. The problem is formulated as an optimal stopping rule problem where it is required to decide at which channel the SU should stop sensing and begin transmission. I provide a closed-form expression for this optimal stopping rule and the optimal transmission power of secondary user (SU).

The average-delay framework is then presented in a single CR channel system with a base station (BS) that schedules the SUs to minimize the average delay while protecting the primary users (PUs) from harmful interference. One of the contributions of the proposed algorithm is its suitability for heterogeneous-channels systems where users with statistically low channel quality suffer worse delay performances. The proposed algorithm guarantees the prespecified delay performance to each SU without violating the PU's interference constraint.

Finally, in the hard-deadline framework, I propose three algorithms that maximize the system's throughput while guaranteeing the required percentage of packets to be transmitted by their deadlines. The proposed algorithms work in heterogeneous systems where the BS is serving different types of users having real-time (RT) data and non-real-time (NRT) data. I show that two of the proposed algorithms have the low complexity where the power policies of both the RT and NRT users are in closed-form expressions and a low-complexity scheduler.

To my Mom and my Dad

and to Radwa and Essam

ACKNOWLEDGMENTS

First and foremost I wish to thank my advisor and mentor, professor Cihan Tepedelenlioğlu for his continuous support of my Ph.D study and related research. His patience, motivation, and immense knowledge are one of the main reasons that helped me in all the time of research and writing of this thesis. I recall that whenever I called for a meeting he never turned me down even if he was busy. Moreover, his technical and non-technical experience and feedback had a great impact on the development of my research.

I would like to thank the National Science Foundation for funding parts of this research under the grants CCF-1117041. In addition, the department of electrical engineering at ASU as well as CORA (Council of Religious Adviors) have funded other parts of this research.

Also, I would like also to thank professor Lei Ying for his support during my PhD. He has given me the opportunity to have a good number of technical discussions with him. Without these discussions, I believe it would have been difficult for me to understand a lot of fundamental concepts in corner-stone topics of my dissertation. In addition, I would like to thank professor Daniel Bliss and professor Oliver Kosut for being part of my committee. I recall a number of helpful discussions with them during my PhD. Their invaluable comments were really helpful. In addition, I would like to thank professor Ahmed K. Sultan, my bachelor's advisor as well as my master's co-advisor, professor Tamer ElBatt, my master's advisor, as well as professor Antonia Papandreou-Suppappola for their help during the critical points of my PhD.

I wish to thank my friends and roommates Ahmad Salim, Ahmad ElMoslimany and Hossam Halawani and my school mates Ruochen Zeng, Xiaofeng Li, Sivaraman Dasarathan, Shahrouz Sharifi, Kousha Kalantari, Adithya Rajan and Abhilash

Sureshbabu. Without them I would not have unforgettable memories. We went through a lot of academic and non-academic discussions, they were really inspirational.

I finish with Egypt, where the most basic source of my life energy resides: my family. I have an amazing family, unique in many ways. Their support has been unconditional all these years; they have given up many things for me to be here; they have cherished with me every great moment and supported me whenever I needed it. My Dad not only supported me financially but also he was there in the difficult moments with his advice and help. My Mom invested a lot of time and energy to support me while she was thousands of miles away. In addition, she made a number of long journeys from Egypt to the US to be beside me during my study. My sister, Radwa, was an inspiring friend and roommate who whose advice was in the right place and time. Last but not least, my nerd little brother, Essam, was very supportive by his help. His advice and technical experience has saved me a lot of time for my computer simulations. A BIG THANK YOU to my wonderful family and I always ask Allah to bless you all.

TABLE OF CONTENTS

APPENDIX

LIST OF TABLES

LIST OF FIGURES

Chapter 1

INTRODUCTION

Cognitive Radio (CR) systems are emerging wireless communication systems that allow efficient spectrum utilization [1]. CRs refer to devices that coexist with the licensed spectrum owners called the primary users (PUs), and that are capable of detecting their presence. Once PU's activity is detected on some frequency channel, the CR user refrains from any further transmission on this channel. This may result in service disconnection for the CR user, thus degrading the quality of service (QoS). If the CR users have access to other channels, the QoS can be improved by switching to another frequency channel instead of completely stopping transmission. If not, then they should control their transmission power to avoid harmful interference to the PUs. Hence, CR users are required to adjust their transmission power levels, and -thus- their rates, according to the interference level the PUs can tolerate. This adjustment could lead to severe degradation in the QoS provided for the CR users, if not designed carefully.

1.1 Cognitive Radio Transmission Schemes

There are two main transmission schemes that CR systems may follow to coexist with the PUs; the overlay and the underlay. In the overlay, CR users, also referred to as the secondary users (SUs), transmit their signal only when the PUs are not using the channel. In other words, the SUs look for the spectrum holes to transmit their data as in Fig. 1.1. Hence, unlike conventional radios, SUs's radios are equipped with a spectrum sensor that is used to sense the spectrum before beginning the transmission phase. In this sensing phase, the SUs listen to all frequency channels to overhear the PUs' transmission so as to decide which channels are free from PUs and which are not. Upon this detection process, the SU picks up a channel, or more, out of

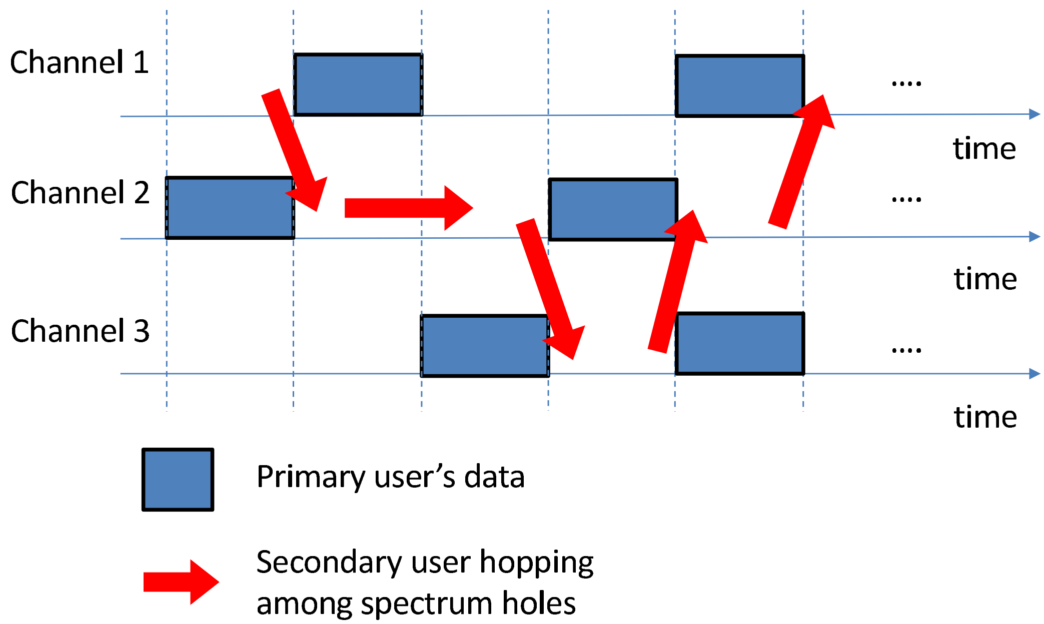

Figure 1.1: Spectrum holes are the locations of the unused spectrum in time and frequency.

the detected-free channels to transmit its data over for a limited amount of time. Once the channel is occupied again by the PU, the SU is expected to refrain from transmission over this channel but allowed to use a different channel after performing the sensing phase again. A practical spectrum sensor might yield wrong decisions, namely, it might detect the presence of a PU on some channel although this channel is actually free, or might miss-detect the PU when it is using the channel. These events are referred to as the false-alarm and miss-detection events, respectively. The higher the false-alarm probability the higher the SU misses transmission opportunities and, thus, the lower the SU's throughput is. Similarly, the higher the probability of miss-detection the more the SU's packet collides with the PU's and leading to a lower throughput since collided packets are lost. While the false-alarm probability affects the SU's throughput alone, the miss-detection probability affects both the SU and the PU. As the sensing phase duration increases, these two probabilities decrease simultaneously. However, increasing the sensing phase duration comes at the expense

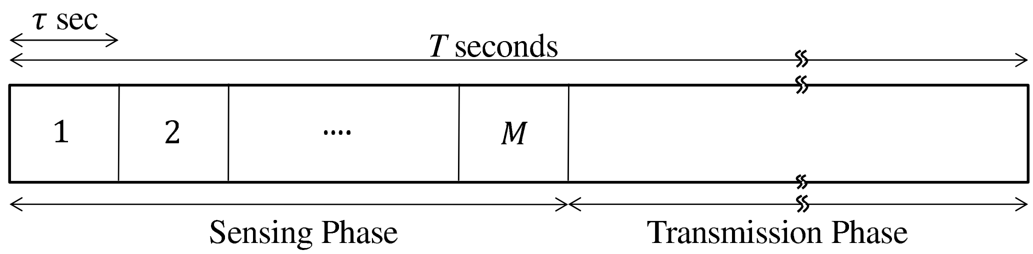

Figure 1.2: The sensing phase is used to sense M channels to detect the presence of the PU. The SU starts transmitting its data in the transmission phase on one of the free channels.

of the transmission phase duration thus decreasing the throughput. This tradeoff has been studied extensively in the literature [2].

In the underlay scheme, the SU is allowed to transmit over any frequency channel at any time as long as the PU can tolerate the interference caused by this transmission. This tolerable level is referred to as the interference temperature as dictated by the Federal Communications Commission (FCC) [3]. In order to guarantee this protection for the PU, the SU has to adjust its transmission power according to the gain of the channel to the primary receiver referred to as the interference channel. The knowledge of this gain instantaneously is essential at the SU's transmitter. While this channel knowledge might be infeasible in CR systems that assume no cooperation between the PU and the SU, in some scenarios the SU might be able to overhear the pilots sent by the primary receiver when it is acting as a transmitter if the PU is using a time division duplex scheme.

In both cases, the overlay and the underlay, the SU might interfere with the PU. This in turn dictates that the SU should adopt its channel access scheme in such a way that this interference is tolerable so that the PU's quality of service (QoS) is not degraded. With that being said, we might expect that the SUs located physically closer to the PUs might suffer larger degradation in their QoS compared to those that are far because closer SUs transmit with smaller amounts of power. This problem does not appear in conventional non CR cellular systems since frequency channels tend to be orthogonal in non CR systems. In other words, in non CR systems, all users are allocated the channels via some scheduler that guarantees those users do not interfere with each other. While in CR systems, since SUs interfere with PUs, we need to develop scheduling and power control algorithms that prevent harmful interference to PUs, as well as guaranteeing acceptable QoS for the SUs.

1.2 Guaranteeing Quality of Service in Cognitive Radio Systems

Since CR users operate in an interference limited environment, they are expected to experience lower QoS than in conventional systems. However, the QoS provided needs to fall within the acceptable level that varies with the application. For example, the average delay of a packet in online streaming is required to be not more than 300ms while that in online gaming should not exceed 50ms. However, these two applications might tolerate small losses in their transmitted packets which is not the case with some other applications as file sharing and email applications that, on the other hand, might tolerate packet delays.

The QoS can include, but is not limited to, throughput, delay, bit-error-rate, interference caused to the PU. Out of these metrics the most two major ones are the throughput and the delay that have gained strong attention in the literature recently [4]. The throughput metric is defined as the average amount of packets (or bits) per channel-use that can be delivered in the SU's network without violating the PU's interference constraints. On the other hand, the delay refers to as the amount of time elapsed from the instant a packet joins the SU's buffer until it is successfully and fully transmitted to its intended receiver. A higher throughput is usually achieved by the efficient power allocation algorithms while better delay performances are usually achieved by efficient scheduling of users.

The problem of scheduling and/or power control has been widely studied in the literature (see [5–11], and references therein). These works aim at optimizing the throughput, providing delay guarantees and/or guaranteeing protection from interference. There are two different frameworks to design scheduling algorithms for real-time packets: the first is referred to in this manuscript as the “average-delay” framework while the second is referred to as the “hard-deadline” framework. The former imposes a bound on the average time a packet spends in the queue before being transmitted to the user. The latter, on the other hand, requires every packet in the system to be transmitted before a pre-specified hard deadline [12]. Clearly, due to the randomness in the arrival process and transmission process, there exists no practical scheduling algorithm that guarantees the transmission of all packets before this hard deadline. In other words, there will exist some packets that will miss their deadlines. Hence, the authors of [13] measure the performance of their algorithm by the percentage of the packets that do not miss their deadlines. The higher this percentage is, the better indication the algorithm is. In this work, we consider the problem of scheduling and power allocation under the average-delay framework as well as the hard-deadline framework.

Before discussing these two frameworks, we study a common factor inherited in both: the “*service time*”. The service time is the amount of time required to transmit a packet from the start of the transmission of its first bit until the transmission of its last bit. The smaller this time is, the better quality a packet will experience under both frameworks.

1.3 Service Time

The service time is affected by the amount of resources allocated to the packet at the time of transmission. Resources might include power, channel bandwidth, coding rate and transmission time. Several works have been done to address how to optimally allocate these resources over time and users. However, from a practical implementation point of view, the most challenging resource is channel bandwidth. This is because increasing the bandwidth requires allocating multiple channels to a user which might require the user to be equipped with high cost transmitters (receivers) capable of transmitting (receiving) over multiple channels simultaneously. On the other hand, allocating a single fixed channel to a user is not optimal.

The problem of channel allocation in multi-channel CR systems has gained attention in recent works due to the challenges associated with the sensing and access mechanisms in a multichannel CR system. Practical hardware constraints on the SUs' transceivers may prevent them from sensing multiple channels simultaneously to detect the state of these channels (free/busy). This leads the SU to sense the channels sequentially, then decide which channel should be used for transmission [14, 15]. In a time slotted system if sequential channel sensing is employed, the SU senses the channels one at a time and stops sensing when a channel is found free. But due to the independent fading among channels, the SU is allowed to skip a free channel if its quality, measured by its power gain, is low and sense another channel seeking the possibility of a higher future gain. Otherwise, if the gain is high, the SU stops at this free channel to begin transmission. The question of when to stop sensing can be formulated as an optimal stopping rule problem [15–18]. In [16] the authors present the optimal stopping rule for this problem in a non-CR system. The work in [15] develops an algorithm to find the optimal order by which channels are to be sequentially sensed in a CR scenario, whereas [17] studies the case where the SUs are allowed to transmit on multiple contiguous channels simultaneously. The authors presented the optimal stopping rule for this problem in a non-fading wireless channel. Transmissions on multiple channels simultaneously may be a strong assumption for low-cost transceivers especially when they cannot sense multiple channels simultaneously.

In general, if a perfect sensing mechanism is adopted, the SU will not cause interference to the PU since the former transmits only on spectrum holes (referred to as an overlay system). Nevertheless, if the sensing mechanism is imperfect, or if the SU's system is an underlay one (where the SU uses the channels as long as the interference to the PU is tolerable), the transmitted power needs to be controlled to prevent harmful interference to the PU. References [19] and [5] consider power control and show that the optimal power control strategy is a water-filling approach under some interference constrain imposed on the SU transmitter. Yet, all of the above work studies single channel systems which cannot be extended to multiple channels in a straightforward manner. A multiuser CR system was considered in [20] in a time slotted system. To allocate the frequency channel to one of the SUs, the authors proposed a contention mechanism that does not depend on the SUs' channel gains, thus neglecting the advantage of multiuser diversity. A major challenge in a multi-channel system is the sequential nature of the sensing where the SU needs to take a decision to stop and begin transmission, or continue sensing based on the information it has so far. This decision needs to trade-off between waiting for a potentially higher throughput and taking advantage of the current free channel. Moreover, if transmission takes place on a given channel, the SU needs to decide the amount of power transmitted to maximize its throughput given some average interference and average power constraints.

In Chapter 2, we model the overlay and underlay scenarios of a multi-channel CR system that are sensed sequentially. The problem is solved for a single SU first then we discuss extensions to a multi-SU scenario. For the single SU case, the problem is formulated as a joint optimal-stopping-rule and power-control problem with the goal of maximizing the SU's throughput subject to average power and average interference constraints. This formulation results in increasing the expected service time of the SU's packets. The expected service time is the average number of time slots that pass while the SU attempts to find a free channel, before successfully transmitting a packet. The increase in the service time is due to skipping free channels, due to their poor gain, hoping to find a future channel of sufficiently high gain. If no channels having a satisfactory gain were found, the SU will not be able to transmit its packet, and will have to wait for longer time to find a satisfactory channel. This increase in service time increases the queuing delay. Thus, we solve the problem subject to a bound on the expected service time which controls the delay. In the multiple SUs case, we show that the solution to the single SU problem can be applied directly to the multi-SU system with a minor modification. We also show that the complexity of the solution decreases when the system has a large number of SUs.

1.4 Average-Delay Framework

The average delay, or simply the delay, is defined as the average amount of time a packet spends in the system starting from the instant it arrives to the buffer until it is completely transmitted. The delay consists of two main factors, the service time, discussed in the previous section, and the queue-waiting time. Unlike the service time, the delay due to queue-waiting time is affected by the scheduling algorithm. The more frequently a user is allocated the channel for transmission, the less its queue-waiting time is, but the more the queue-waiting times for the other users are.

Under the average-delay framework, the scheduling algorithm should guarantee that the average queuing delay for each user does not exceed this pre-specified bound. Delay due to the queue-waiting time is also well studied recently in the literature and scheduling algorithms have been proposed to guarantee small delay for users in conventional systems [21–23]. In [21], the authors study the joint scheduling-and-power-allocation problem in the presence of an average power constraint. Although in [21] the proposed algorithm offers an acceptable delay performance, all users are assumed to transmit with the same power. A power allocation and routing algorithm is proposed in [23] to maximize the capacity region under an instantaneous power constraint. While the authors show an upper bound on the average delay, this delay performance is not guaranteed to be optimal.

Although queuing theory, that was originally developed to model packets at a server, can be applied to wireless channels, the challenges are different. One of the main challenges is the fading nature of the wireless channel that changes from a slot to another. Fading requires adapting the user's power and/or rate according to the channel's fading coefficient. The idea of power and/or rate adaptation based on the channel condition does not have an analogy in server problems and, thus, is absent in the aforementioned references. Instead, existing works treat wireless channels as on-off fading channels and do not consider multiple fading levels. Among the relevant references that consider a more general fading channel model are [23], which was discussed above, [24, 25] where the optimization over the scheduling algorithm was out of the scope of their work, and [26] that neglects the interference constraint since it considers a non CR system.

In contrast with [6–9, 27] that do not optimize the queuing delay, the problem of minimizing the sum of SUs' average delays is considered in this work. The proposed algorithm guarantees a bound on the instantaneous interference to the PUs, a guarantee that is absent in [21, 23]. Based on Lyapunov optimization techniques [21], an algorithm that dynamically schedules the SUs as well as optimally controlling their transmission power is presented.

1.5 Hard-Deadline Framework

While the average-delay framework might perform well for online streaming of pre-recorded audio/video files, its performance in online streaming of on-air broadcast data such as video conference calls is questionable. This is because, unlike pre-recorded data of a finite time length, video calls have an endless stream of data that needs to arrive in a timely manner. Moreover, not all packets have to be delivered to the end user to have an acceptable QoS for a video call. Hence, the authors of [13] present the second framework for modeling real-time traffic over wireless networks, namely, the "hard-deadline" framework. Quality-of-service-based scheduling has received attention recently. It is shown in [28] and [29] that quality-of-service-aware scheduling results in a better performance in LTE systems compared to quality-of-service-unaware best-effort techniques. Depending on the application, quality-of-service (QoS) metrics may refer to long-term throughput [30], short-term throughput [31], per-user average delay [32], average number of packets missing a specific deadline [13], or the average time a user waits to receive its data [33]. Real-time audio and video applications need to be served by algorithms that take hard deadlines into consideration. That is, these algorithms need to be aware that these packets have to be served before a certain deadline passes. This is because if a real-time packet is not transmitted on time, the corresponding user might experience intermittent connectivity of its audio or video.

In [13] the authors consider binary erasure channels and present a sufficient and necessary condition to determine if a given problem is feasible. The work is extended in three different directions. The first direction studies the problem under delayed feedback [29]. The second considers general channel fading models [34]. While the third studies multicast video packets that have strict deadlines and utilize network coding to improve the overall network performance [35, 36]. Unlike the time-framed assumption in the previous works, the authors of [11] assume that arrivals and deadlines do not have to occur at the edges of a time frame. They present a scheduling algorithm under the on-off channel fading model and present its achievable region under general arrivals and deadline patterns but with a fixed power transmission. In [37] the authors study the scheduling problem in the presence of real-time and non-real-time data. Unlike real-time data, non-real-time data do not have strict deadlines but have an implicit stability constraint on the queues. Using the dual function approach, the problem was decomposed into an online algorithm that guarantees network stability and real-time users' satisfaction.

Power allocation has not been considered for RT users in the literature, to the best of our knowledge. In this chapter, we study resource allocation in the presence of simultaneous RT and NRT users in a downlink cellular system. We formulate the problem as a joint scheduling-and-power-allocation problem to maximize the sum throughput of the NRT users subject to an average power constraint on the base station (BS), as well as a delivery ratio requirement constraint for each RT user. The delivery ratio constraint requires a minimum ratio of packets to be transmitted by a hard deadline, for each RT user. Perhaps the closest to this work are references [37] and [27]. The former does not consider power allocation, while the latter assumes that only one user can be scheduled per time slot.

Chapter 2

DELAY DUE TO SERVICE TIME

In this chapter we study the delay resulting from the service time of packets and neglect the delay resulting from the waiting time in the queues. We treat the cognitive radio system as a single secondary user (SU) accessing a multi-channel system. The main problem studied in this chapter is the tradeoff between the service time and the throughput. We assume the SU senses the channels sequentially to detect the presence of the primary user (PU), and stops its search to access a channel if it offers a significantly high throughput. The tradeoff exists because stopping at early-sensed channels gives low average service time but, at the same time, gives low throughput since early channels might have low gains. The joint optimal stopping rule and power control problem is formulated as a throughput maximization problem with an average service time and power constraint. We note that in this chapter we use the word delay to refer to the service time.

To the best of our knowledge, this is the first work to study the joint power-control and optimal-stopping-rule problem in a multi channel CR system. The contribution in this chapter is the formulation of a joint power-control and optimal-stopping-rule problem that also incorporates a delay constraint and present a low complexity solution in the presence of interference/collision constraint from the SU to the PU due to the imperfect sensing mechanism. The preliminary results in [38] consider an overlay framework for single user case while neglecting sensing errors. But in this work, we also study the problem in the underlay scenario where interference is allowed from the secondary transmitter (ST) to the primary receiver (PR) and extend to multiple SU case. We also generalize the solution to the multi-SU case when the number of SUs is large. We discuss the applicability of our formulation in typical delay-constrained scenarios where packets arrive simultaneously and have a same deadline. We show that the proposed algorithm can be used to solve this problem offline, to maximize the throughput and meet the deadline constraint at the same time. Moreover, we propose an online algorithm that gives higher throughput compared to the offline approach while meeting the deadline constraint.

2.1 Overlay System Model

Consider a PU network that has a licensed access to M orthogonal frequency channels. Time is slotted with a time slot duration of T seconds. The SU's network consists of a single ST (SU and ST will be used interchangeably) attempting to send real-time data to its intended secondary receiver (SR) through one of the channels licensed to the PU. Before a time slot begins, the SU is assumed to have ordered the channels according to some sequence (we note that the method of ordering the channels is outside the scope of this work. The reader is referred to [15] for further details about channel ordering), labeled $1, ..., M$. The set of channels is denoted by $\mathcal{M} = \{1, ..., M\}$. Before the SU attempts to transmit its packet over channel i, it senses this channel to determine its availability "state" which is described by a Bernoulli random variable b_i with parameter θ_i (θ_i is called the availability probability of channel i). If $b_i = 0$ (which happens with probability θ_i), then channel i is free and the SU may transmit over it until the on-going time slot ends. If $b_i = 1$, channel i is busy, and the SU proceeds to sense channel $i + 1$. Channel availabilities are statistically independent across frequency channels and across time slots.

We assume that the SU has limited capabilities in the sense that no two channels can be sensed simultaneously. This may be the case when considering radios having a single sensing module with a fixed bandwidth, so that it can be tuned to only one frequency channel at a time. The reader is referred to [39], [40] and [41]

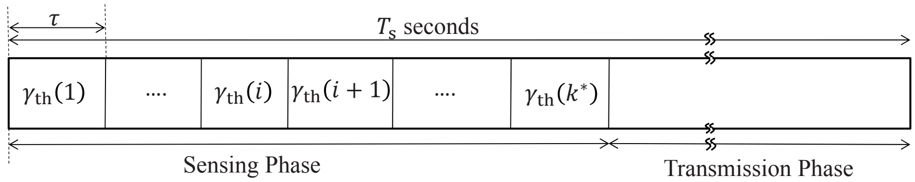

Figure 2.1: Sensing and transmission phases in one time slot. The SU senses each channel for τ seconds, determines its state, then probes the gain if the channel is found free. The sensing phase ends if the probed gain $\gamma_i > \gamma_{th}(i)$, in which case $k^* = i$. Hence, k^* is a random variable that depends on the channel states and gains.

for detailed information on advanced spectrum sensing techniques. Therefore, at the beginning of a given time slot, the SU selects a channel, say channel 1, senses it for τ seconds ($\tau \ll T/M$), and if it is free, the SU transmits on this channel if its channel gain is high enough[1]. Otherwise, the SU skips this channel and senses channel 2, and so on until it finds a free channel. If all channels are busy (i.e. the PU has transmission activities on all M channels) then this time slot will be considered as "blocked". In this case, the SU waits for the following time slot and begins sensing following the same channel sensing sequence. As the sensing duration increases, the transmission phase duration decreases which decreases the throughput. But we cannot arbitrarily decrease the value of τ since this decreases the reliability of the sensing outcome. This trade-off has been studied extensively in the literature, e.g. [42], [43]. In this work we study the impact of sequential channel sensing on the throughput rather than the sensing duration on the throughput. Hence we assume that τ is a fixed parameter and is not optimized over. For details on the trade-off between throughput and sensing duration in this sequential sensing problem the reader is referred to [2].

The fading channel between ST and SR is assumed to be flat fading with independent, identically distributed (i.i.d.) channel gains across the M channels. To achieve higher data rates, the SU adapts its data rate according to the instantaneous

[1] How "high" is "high" is going to be explained later

power gain of the channel before beginning transmission on this channel. To do this, once the SU finds a free channel, say channel i, the gain γ_i is probed. The data rate will be proportional to $\log(1 + P_{1,i}(\gamma_i)\gamma_i)$, where $P_{1,i}(\gamma_i)$ is the power transmitted by the SU at channel i as a function of the instantaneous gain [44]. Fig. 2.1 shows a potential scenario where the SU senses k^* channels, skips the first $k^* - 1$, and uses the k^*th channel for transmission until the end of this on-going time slot. In this scenario the SU "stops" at the k^*th channel, for some $k^* \in \mathcal{M}$. Stopping at channel i depends on two factors: 1) the availability of channel b_i, and 2) the instantaneous channel gain γ_i. Clearly, b_i and γ_i are random variables that change from one time slot to another. Hence, k^*, that depends on these two factors, is a random variable. More specifically, it depends on the states $[b_1, ..., b_M]$ along with the gains of each channel $[\gamma_1, ..., \gamma_M]$. To understand why, consider that the SU senses channel i, finds it free and probes its gain γ_i. If γ_i is found to be low, then the SU skips channel i (although free) and senses channel $i + 1$. This is to take advantage of the possibility that $\gamma_j \gg \gamma_i$ for $j > i$. On the other hand, if γ_i is sufficiently large, the SU stops at channel i and begins transmission. In that latter case $k^* = i$. Defining the two random vectors $\underline{b} = [b_1, ..., b_M]^T$ and $\underline{\gamma} = [\gamma_1, ..., \gamma_M]^T$, k^* is a deterministic function of $\underline{b}$ and $\underline{\gamma}$.

We define the stopping rule by defining a threshold $\gamma_{\text{th}}(i)$ to which each γ_i is compared when the ith channel is found free. If $\gamma_i \geq \gamma_{\text{th}}(i)$, channel i is considered to have a "high" gain and hence the SU "stops" and transmits at channel i. Otherwise, channel i is skipped and channel $i + 1$ sensed. In the extreme case when $\gamma_{\text{th}}(i) = 0$, the SU will not skip channel i if it is found free. Increasing $\gamma_{\text{th}}(i)$ allows the SU to skip channel i whenever $\gamma_i < \gamma_{\text{th}}(i)$, to search for a better channel, thus potentially increasing the throughput. Setting $\gamma_{\text{th}}(i)$ too large allows channel i to be skipped even if γ_i is high. This constitutes the trade-off in choosing the thresholds $\gamma_{\text{th}}(i)$.

The optimal values of $\gamma_{\text{th}}(i)$ $i \in \mathcal{M}$, determine the optimal stopping rule.

Let $P_{1,i}(\gamma)$ denote the power transmitted at the ith channel when the instantaneous channel gain is γ, if channel i was chosen for transmission. Since the SU can transmit on one channel at a time, the power transmitted at any time slot at channel i is $P_{1,i}(\gamma_i)\mathbb{1}\,(i = k^*)$, where $\mathbb{1}\,(i = k^*) = 1$ if $i = k^*$ and 0 otherwise. Define $c_i \triangleq 1 - \frac{i\tau}{T}$ as the fraction of the time slot remaining for the SU's transmission if the SU transmits on the ith channel in the sensing sequence. The average power constraint is $\mathbb{E}_{\underline{\gamma},\underline{b}}[c_{k^*}P_{k^*}(\gamma_{k^*})] \leq P_{\text{avg}}$, where the expectation is with respect to the random vectors $\underline{\gamma}$ and $\underline{b}$. We will henceforth drop the subscript from the expected value operator $\mathbb{E}$. This expectation can be calculated recursively from

$$S_i(\mathbf{\Gamma}_{\text{th}}(i), \mathbf{P}_{1,i}) = \theta_i c_i \int_{\gamma_{\text{th}}(i)}^{\infty} P_{1,i}(\gamma) f_{\gamma_i}(\gamma)\, d\gamma + \left[1 - \theta_i \bar{F}_{\gamma_i}(\gamma_{\text{th}}(i))\right] S_{i+1}(\mathbf{\Gamma}_{\text{th}}(i+1), \mathbf{P}_{i+1}), \tag{2.1}$$

$i \in \mathcal{M}$, where $\mathbf{P}_{1,i} \triangleq [P_{1,i}(\gamma), ..., P_{1,M}(\gamma)]^T$ and $\mathbf{\Gamma}_{\text{th}}(i) \triangleq [\gamma_{\text{th}}(i), ..., \gamma_{\text{th}}(M)]^T$ are the vectors of the power functions and thresholds respectively, with $S_{M+1}(\mathbf{\Gamma}_{\text{th}}(M+1), \mathbf{P}_{M+1}) \triangleq 0$, $f_{\gamma_i}(\gamma)$ is the Probability Density Function (PDF) of the gain γ_i of channel i, and $\bar{F}_{\gamma_i}(x) \triangleq \int_x^{\infty} f_{\gamma_i}(\gamma)\, d\gamma$ is the complementary cumulative distribution function. The first term in (2.1) is the average power transmitted at channel i given that channel is chosen for transmission (i.e. given that $k^* = i$). The second term represents the case where channel i is skipped and channel $i+1$ is sensed. It can be shown that $S_1(\mathbf{\Gamma}_{\text{th}}(1), \mathbf{P}_{1,1}) = \mathbb{E}\left[c_{k^*}P_{k^*}(\gamma)\right]$. Moreover, we will also drop the index i from the subscript of $f_{\gamma_i}(\gamma)$ and $\bar{F}_{\gamma_i}(\gamma)$ since channels suffer i.i.d. fading. Although we have only included an average power constraint in our problem, we will modify, after solving the problem, the solution to include an instantaneous power constraint as well.

The SU's average throughput is defined as $\mathbb{E}[c_{k^*}\log(1 + P_{k^*}(\gamma_{k^*})\gamma_{k^*})]$. Similar to the average power, we denote the expected throughput as $U_1(\mathbf{\Gamma}_{\text{th}}(1), \mathbf{P}_{1,1})$ which can be derived using the following recursive formula

$$U_i(\mathbf{\Gamma}_{\text{th}}(i),\mathbf{P}_{1,i}) = \theta_i c_i \int_{\gamma_{\text{th}}(i)}^{\infty} \log\left(1 + P_{1,i}(\gamma)\gamma\right) f_\gamma(\gamma)\, d\gamma +$$
$$\left[1 - \theta_i \bar{F}_\gamma(\gamma_{\text{th}}(i))\right] U_{i+1}\left(\mathbf{\Gamma}_{\text{th}}(i+1), \mathbf{P}_{i+1}\right) \tag{2.2}$$

$i \in \mathcal{M}$, with $U_{M+1}(\cdot,\cdot) \triangleq 0$. $U_1(\mathbf{\Gamma}_{\text{th}}(1), \mathbf{P}_{1,1})$ represents the expected data rate of the SU as a function of the threshold vector $\mathbf{\Gamma}_{\text{th}}(1)$ and the power function vector $\mathbf{P}_{1,1}$.

If the SU skips all channels, either due to being busy, due to their low gain or due to a combination of both, then the current time slot is said to be blocked. The SU has to wait for the following time slot to begin searching for a free channel again. This results in a delay in serving (transmitting) the SU's packet. Define the delay D as the number of time slots the SU consumes before successfully transmitting a packet. That is, $D - 1$ is a random variable that represents the number of consecutively blocked time slots. In real-time applications, there may exist some average delay requirement $\bar{D}_{\max}$ on the packets that must not be exceeded. Since the availability of each channel is independent across time slots, D follows a geometric distribution having $\mathbb{E}[D] = (\Pr[\text{Success}])^{-1}$ where $\Pr[\text{Success}] = 1 - \Pr[\text{Blocking}]$. In other words, $\Pr[\text{Success}]$ is the probability that the SU finds a free channel with high enough gain so that it does not skip all M channels in a time slot. It is given by $\Pr[\text{Success}] \triangleq p_1(\mathbf{\Gamma}_{\text{th}}(1))$ which can be calculated recursively using the following equation

$$p_i(\mathbf{\Gamma}_{\text{th}}(i)) = \theta_i \bar{F}_\gamma(\gamma_{\text{th}}(i)) + \left[1 - \theta_i \bar{F}_\gamma(\gamma_{\text{th}}(i))\right] p_{i+1}(\mathbf{\Gamma}_{\text{th}}(i+1)), \tag{2.3}$$

$i \in \mathcal{M}$, where $p_{M+1} \triangleq 0$. Here, $p_i(\mathbf{\Gamma}_{\text{th}}(i))$ is the probability of transmission on channel i, $i+1$,..., or M.

2.2 Problem Statement and Proposed Solution

From equation (2.2) we see that the SU's expected throughput U_1 depends on the threshold vector $\mathbf{\Gamma}_{\text{th}}(1)$ and the power vector $\mathbf{P}_{1,1}$. The goal is to find the optimum values of $\mathbf{\Gamma}_{\text{th}}(1) \in \mathbb{R}^M$ and functions $\mathbf{P}_{1,1}$ that maximize U_1 subject to an average power constraint and an expected packet delay constraint. The delay constraint can be written as $\mathbb{E}[D] \leq \bar{D}_{\max}$ or, equivalently, $p_1(\mathbf{\Gamma}_{\text{th}}(1)) \geq 1/\bar{D}_{\max}$. Mathematically, the problem becomes

$$
\begin{array}{ll}
\text{maximize} & U_1(\mathbf{\Gamma}_{\text{th}}(1), \mathbf{P}_{1,1}) \\
\text{subject to} & S_1(\mathbf{\Gamma}_{\text{th}}(1), \mathbf{P}_{1,1}) \leq P_{\text{avg}} \\
& p_1(\mathbf{\Gamma}_{\text{th}}(1)) \geq \frac{1}{\bar{D}_{\max}} \\
\text{variables} & \mathbf{\Gamma}_{\text{th}}(1), \mathbf{P}_{1,1},
\end{array}
\tag{2.4}
$$

where the first constraint represents the average power constraint, while the second is a bound on the average packet delay. We allow the power $P_{1,i}$ to be an arbitrary function of γ_i and optimize over this function to maximize the throughput subject to average power and delay constraints. Even though (2.4) is not proven to be convex, we provide closed-form expressions for the optimal thresholds and power-functions vector. To this end, we first calculate the Lagrangian associated with (2.4). Let λ_{P} and λ_{D} be the dual variables associated with the constraints in problem (2.4). The Lagrangian for (2.4) becomes

$$
\begin{aligned}
&L\left(\mathbf{\Gamma}_{\text{th}}(1), \mathbf{P}_{1,1}, \lambda_{\text{P}}, \lambda_{\text{D}}\right) = U_1\left(\mathbf{\Gamma}_{\text{th}}(1), \mathbf{P}_{1,1}\right) - \\
&\quad \lambda_{\text{P}}\left(S_1(\mathbf{\Gamma}_{\text{th}}(1), \mathbf{P}_{1,1}) - P_{\text{avg}}\right) + \lambda_{\text{D}}\left(p_1(\mathbf{\Gamma}_{\text{th}}(1)) - \frac{1}{\bar{D}_{\max}}\right).
\end{aligned}
\tag{2.5}
$$

Differentiating (2.5) with respect to each of the primal variables $P_{1,i}(\gamma)$ and $\gamma_{\text{th}}(i)$ and equating the resulting derivatives to zero, we obtain the KKT equations below which are necessary conditions for optimality [45, 46]:

$$P_{1,i}^{*}(\gamma)=\left(\frac{1}{\lambda_{\mathrm{P}}^{*}}-\frac{1}{\gamma}\right)^{+}, \quad \gamma>\gamma_{\mathrm{th}}^{*}(i), \tag{2.6}$$

$$\begin{aligned}&\log\left(1+\left(\frac{1}{\lambda_{\mathrm{P}}^{*}}-\frac{1}{\gamma_{\mathrm{th}}^{*}(i)}\right)^{+}\gamma_{\mathrm{th}}^{*}(i)\right)-\lambda_{\mathrm{P}}^{*}\left(\frac{1}{\lambda_{\mathrm{P}}^{*}}-\frac{1}{\gamma_{\mathrm{th}}^{*}(i)}\right)^{+}\\&=\frac{U_{i+1}^{*}-\lambda_{\mathrm{P}}^{*}S_{i+1}^{*}-\lambda_{\mathrm{D}}^{*}\cdot\left(1-p_{i+1}^{*}\right)}{c_i},\end{aligned} \tag{2.7}$$

$$S_1^{*}\leq P_{\mathrm{avg}}\ ,\ p_1^{*}\geq\frac{1}{\bar{D}_{\max}}\ ,\ \lambda_{\mathrm{P}}^{*}\geq 0\ ,\ \lambda_{\mathrm{D}}^{*}\geq 0, \tag{2.8}$$

$$\lambda_{\mathrm{P}}^{*}\cdot\left(S_1^{*}-P_{\mathrm{avg}}\right)=0, \tag{2.9}$$

$$\lambda_{\mathrm{D}}^{*}\cdot\left(p_1^{*}-\frac{1}{\bar{D}_{\max}}\right)=0, \tag{2.10}$$

$i\in\mathcal{M}$. We use $U_{i+1}^{*}\triangleq U_{i+1}\left(\boldsymbol{\Gamma}_{\mathrm{th}}^{*}(i+1),\mathbf{P}_{i+1}^{*}\right)$ while $S_{i+1}^{*}\triangleq S_{i+1}\left(\boldsymbol{\Gamma}_{\mathrm{th}}^{*}(i+1),\mathbf{P}_{i+1}^{*}\right)$ and $p_{i+1}^{*}\triangleq p_{i+1}\left(\boldsymbol{\Gamma}_{\mathrm{th}}^{*}(i+1)\right)$ for brevity in the sequel. We note that $U_{M+1}(\cdot,\cdot)=S_{M+1}(\cdot,\cdot)=p_{M+1}(\cdot)\triangleq 0$ by definition. We observe that these KKT equations involve the primal ($\boldsymbol{\Gamma}_{\mathrm{th}}^{*}(1)$ and $\mathbf{P}_1^{*}$) and the dual (λ_{P}^{*} and λ_{D}^{*}) variables. Our approach is to find a closed-form expression for the primal variables in terms of the dual variables, then propose a low-complexity algorithm to obtain the solution for the dual variables. The optimality of this approach is discussed at the end of this section (in Section 2.2.3) where we show that, loosely speaking, the KKT equations provide a unique solution to the primal-dual variables. Hence, based on this unique solution, and on the fact that the KKT equations are necessary conditions for the optimal solution, then this solution is not only necessary but sufficient as well, and hence optimal.

2.2.1 Solving for Primal Variables

Equation (2.6) is a water-filling strategy with a slight modification due to having the condition $\gamma>\gamma_{\mathrm{th}}(i)$. This condition comes from the sequential sensing of the channels which is absent in the classic water-filling strategy [44]. Equation (2.6) gives a closed-form solution for $\mathbf{P}_{1,1}$. On the other hand, the entries of the vector $\boldsymbol{\Gamma}_{\mathrm{th}}^{*}(1)$

are found via the set of equations (2.7). Note that equation (2.7) indeed forms a set of M equations, each solves for one of the $\gamma^*_{\text{th}}(i)$, $i \in \mathcal{M}$. We refer to this set as the "threshold-finding" equations. For a given value of i, solving for $\gamma^*_{\text{th}}(i)$ requires the knowledge of only $\gamma^*_{\text{th}}(i+1)$ through $\gamma^*_{\text{th}}(M)$, and does not require knowing $\gamma^*_{\text{th}}(1)$ through $\gamma^*_{\text{th}}(i-1)$. Thus, these M equations can be solved using back-substitution starting from $\gamma^*_{\text{th}}(M)$. To solve for $\gamma^*_{\text{th}}(i)$, we use the fact that $\gamma^*_{\text{th}}(i) \geq \lambda^*_{\text{P}}$ that is proven in the following lemma.

Lemma 1. *The optimal solution of problem (2.4) satisfies* $\gamma^*_{th}(i) \geq \lambda^*_{\text{P}}$ $\forall i \in \mathcal{M}$.

Proof. See Appendix A for proof. □

The intuition behind Lemma 1 is as follows. If, for some channel i, $\gamma^*_{\text{th}}(i) < \lambda^*_{\text{P}}$ was possible, and the instantaneous gain γ_i happened to fall in the range $[\gamma^*_{\text{th}}(i), \lambda^*_{\text{P}})$ at a given time slot, then the SU will not skip channel i since $\gamma_i > \gamma^*_{\text{th}}(i)$. But the power transmitted on channel i is $P_{1,i}(\gamma_i) = (1/\lambda^*_{\text{P}} - 1/\gamma_i)^+ = 0$ since $\gamma_i < \lambda^*_{\text{P}}$. This means that the SU will neither skip nor transmit on channel i, which does not make sense from the SU's throughput perspective. To overcome this event, the SU needs to set $\gamma^*_{\text{th}}(i)$ at least as large as λ^*_{P} so that whenever $\gamma_i < \lambda^*_{\text{P}}$, the SU skips channel i rather than transmitting with zero power.

Lemma 1 allows us to remove the $(\cdot)^+$ sign in equation (2.7) when solving for $\gamma^*_{\text{th}}(i)$. Rewriting (2.7) we get

$$\begin{aligned}&\frac{-\lambda^*_{\text{P}}}{\gamma^*_{\text{th}}(i)} \exp\left(\frac{-\lambda^*_{\text{P}}}{\gamma^*_{\text{th}}(i)}\right) = \\ &-\exp\left(-\frac{U^*_{i+1} - \lambda^*_{\text{P}} S^*_{i+1} - \lambda^*_{\text{D}} \cdot \left(1 - p^*_{i+1}\right)}{c_i} - 1\right),\ i \in \mathcal{M}, \end{aligned} \tag{2.11}$$

Equation (2.11) is now on the form $W \exp(W) = c$, whose solution is $W = W_0(c)$, where $W_0(x)$ is the principle branch of the Lambert W function [47] and is given by $W_0(x) = \sum_{n=1}^{\infty} \frac{(-n)^{n-1}}{n!} x^n$. The only solution to (2.11) which satisfies Lemma 1 is given for $i \in \mathcal{M}$ by

$$\gamma_{\text{th}}^{*}(i) = \frac{-\lambda_{\text{P}}^{*}}{W_0\left(-\exp\left(-\frac{\left(U_{i+1}^{*}-\lambda_{\text{P}}^{*} S_{i+1}^{*}-\lambda_{\text{D}}^{*}\left(1-p_{i+1}^{*}\right)\right)^{+}}{c_i}-1\right)\right)}. \tag{2.12}$$

Hence, $\mathbf{\Gamma}_{\text{th}}^{*}(1)$ and $\mathbf{P}_1^{*}$ are found via equations (2.12) and (2.6) respectively which are one-to-one mappings from the dual variables $(\lambda_{\text{P}}^{*}, \lambda_{\text{D}}^{*})$. And if we had an instantaneous power constraint $P_{1,i}(\gamma) \leq P_{\max}$, we could write down the Lagrangian and solve for $P_{1,i}(\gamma)$. The details are similar to the case without an instantaneous power constraint and are, thus, omitted for brevity. The reader is referred to [5] for a similar proof. The expression for $P_{1,i}^{*}(\gamma)$ is given by

$$P_{1,i}^{*}(\gamma) = \begin{cases} \left(\frac{1}{\lambda_{\text{P}}^{*}} - \frac{1}{\gamma}\right)^{+} & \text{if } \frac{1}{\lambda_{\text{P}}^{*}} - \frac{1}{\gamma} < P_{max} \\ P_{max} & \text{otherwise.} \end{cases} \tag{2.13}$$

Since the optimal primal variables are explicit functions of the optimal dual variables, once the optimal dual variables are found, the optimal primal variables are found and the optimization problem is solved. We now discuss how to solve for these dual variables.

2.2.2 Solving for Dual Variables

The optimum dual variable λ_{P}^{*} must satisfy equation (2.9). Thus if $\lambda_{\text{P}}^{*} > 0$, then we need $S_1^{*} - P_{\text{avg}} = 0$. This equation can be solved using any suitable root-finding algorithm. Hence, we propose Algorithm 1 that uses bisection [48]. In each iteration n, the algorithm calculates S_1^{*} given that $\lambda_{\text{P}} = \lambda_{\text{P}}^{(n)}$, and given some fixed λ_{D}, compares it to P_{avg} to update $\lambda_{\text{P}}^{(n+1)}$ accordingly. The algorithm terminates when $S_1^{*} = P_{\text{avg}}$, i.e. $\lambda_{\text{P}}^{(n)} = \lambda_{\text{P}}^{*}$. The superiority of this algorithm over the exhaustive search is due to the use of the bisection algorithm that does not go over all the search space of λ_{P}.

In order for the bisection to converge, there must exist a single solution for equation $S_1^* = P_{\text{avg}}$. This is proven in Theorem 1.

Theorem 1. *S_1^* is decreasing in $\lambda_{\text{P}}^* \in [0, \infty)$ given some fixed $\lambda_{\text{D}}^* \geq 0$. Moreover, the optimal value λ_{P}^* satisfying $S_1^* = P_{\text{avg}}$ is upper bounded by $\lambda_{\text{P}}^{\max} \triangleq \sum_{i=1}^{M} \theta_i c_i / P_{\text{avg}}$.*

Proof. See Appendix B for the proof. □

We note that Algorithm 1 can be systematically modified to call any other root-finding algorithm (e.g. the secant algorithm [48] that converges faster than the bisection algorithm).

Algorithm 1 Finding λ_{P}^* given some λ_{D}

Initialize $n \leftarrow 1$, $\lambda_{\text{P}}^{\min} \leftarrow 0$, $\lambda_{\text{P}}^{\max} \leftarrow \sum_{i=1}^{M} \theta_i c_i / P_{\text{avg}}$, $\lambda_{\text{P}}^{(1)} \leftarrow \left(\lambda_{\text{P}}^{\min} + \lambda_{\text{P}}^{\max}\right)/2$
while $|S_1^* - P_{avg}| > \epsilon$ **do**
  Calculate S_1^* given that $\lambda_{\text{P}}^* = \lambda_{\text{P}}^{(n)}$. Call it $S^{(n)}$.
  if $S^{(n)} - P_{\text{avg}} > 0$ **then**
    $\lambda_{\text{P}}^{\min} = \lambda_{\text{P}}^{(n)}$
  else
    $\lambda_{\text{P}}^{\max} = \lambda_{\text{P}}^{(n)}$
  end if
  $\lambda_{\text{P}}^{(n+1)} \leftarrow \left(\lambda_{\text{P}}^{\min} + \lambda_{\text{P}}^{\max}\right)/2$
  $n \leftarrow n + 1$
end while
$\lambda_{\text{P}}^* \leftarrow \lambda_{\text{P}}^{(n)}$

Now, to search for λ_{D}^*, we state the following lemma.

Lemma 2. *The optimum value λ_{D}^* that solves problem (2.4) satisfies $0 \leq \lambda_{\text{D}}^* < \lambda_{\text{D}}^{\max}$, where*

$$\lambda_{\text{D}}^{\max} \triangleq \frac{c_1 \left[\log(t) - t + 1\right] + U_2^{\max}}{1 - p_2^{\max}} \tag{2.14}$$

with $t \triangleq \left(\min\left(\lambda_{\text{P}}^{\max}, \bar{F}_\gamma^{-1}\left(\frac{1}{\theta_1 \bar{D}_{\max}}\right)\right)\right) / \left(\bar{F}_\gamma^{-1}\left(\frac{1}{\theta_1 \bar{D}_{\max}}\right)\right)$ and $U_2^{\max}$ is an upper bound on U_2^* and is given by $\left(\int_{\lambda_{\text{P}}^{\max}}^{\infty} \log\left(\gamma/\lambda_{\text{P}}^{\max}\right) f_\gamma(\gamma)\, d\gamma\right) \left(\sum_{i=2}^{M} \theta_i c_i\right)$, while $p_2^{\max}$ is an upper bound on p_2^* and is given by $\sum_{i=2}^{M} \prod_{j=2}^{i-1} (1 - \theta_j)\, \theta_i$.

Proof. See Appendix C. □

Lemma 2 gives an upper bound on λ_{D}^*. This bound decreases the search space of λ_{D}^* drastically instead of searching over $\mathbb{R}$. Thus the solution of problem (2.4) can be summarized on 3 steps: 1) Fix $\lambda_{\mathrm{D}}^* \in [0, \lambda_{\mathrm{D}}^{\max})$ and find the corresponding optimum λ_{P}^* using Algorithm 1. 2) Substitute the pair $(\lambda_{\mathrm{P}}^*, \lambda_{\mathrm{D}}^*)$ in equations (2.6) and (2.12) to get the power and threshold functions, then evaluate U_1^* from (2.2). 3) Repeat steps 1 and 2 for other values of λ_{D}^* until reaching the optimum λ_{D}^* that satisfies $p_1^* = 1/\bar{D}_{\max}$. If there are multiple λ_{D}^*'s satisfying $p_1^* = 1/\bar{D}_{\max}$, then the optimum one is the one that gives the highest U_1^*.

Although the order by which the channels are sensed is assumed fixed, the proposed algorithm can be modified to optimize over the sensing order by a relatively low complexity sorting algorithm. Particularly, the dynamic programming proposed in [15] can be called by Algorithm 1 to order the channels. The complexity of the sorting algorithm alone is $O(2^M)$ compared to the $O(M!)$ of the exhaustive search to sort the M channels. The modification to our proposed algorithm would be in step 3 of Algorithm 1, where S_1^* would be optimized over the number of channels (as well as $\mathbf{\Gamma}_{\mathrm{th}}^*(1)$).

2.2.3 Optimality of the Proposed Solution

Since the problem in (2.4) is not proven to be convex, the KKT conditions provide only necessary conditions for optimality and need not be sufficient [49]. This means that there might exist multiple solutions (i.e. multiple solutions for the primal and/or dual variables) satisfying the KKT conditions, at least one of which is optimal. But since Theorem 1 proves that there exists one unique solution to λ_{P}^* given λ_{D}^*, then $\mathbf{\Gamma}_{\mathrm{th}}^*(1)$ and $\mathbf{P}_1^*$ are unique as well (from equations (2.6) and (2.12)) given some λ_{D}^*. Hence, by sweeping λ_{D}^* over $[0, \lambda_{\mathrm{D}}^{\max})$, we have a unique solution satisfying the KKT conditions, which means that the KKT conditions are sufficient as well and our approach is optimal for problem (2.4).

2.3 Generalization of Deadline Constraints

In the overlay and underlay schemes discussed thus far, we were assuming that each packet has a hard deadline of one time slot. If a packet is not delivered as soon as it arrives at the ST, then it is dropped from the system. But in real-time applications, data arrives at the ST's buffer on a frame-by-frame structure. Meaning multiple packets (constituting the same frame) arrive simultaneously rather than one at a time. A frame consists of a fixed number of packets, and each packet fits into exactly one time slot of duration T. Each frame has its own deadline and thus, packets belonging to the same frame have the same deadline [36]. This deadline represents the maximum number of time slots that the packets, belonging to the same frame, need to be transmitted by.

In this section we solve this problem for the overlay scenario. The solution presented in Section 2.2 can be thought of as a special case of the problem presented in this section where the deadline was equal to 1 time slot and each frame consists of one packet. We show that the solution presented in Section 2.2 can be used to solve this generalized problem in an offline fashion (i.e. before attempting to transmit any packet of the frame). Moreover, we propose an online update algorithm that updates the thresholds and power functions each time slot and show that this outperforms the offline solution.

2.3.1 Offline Solution

Assume that each frame consists of K packets and that the entire frame has a deadline of t_f time slots ($t_f > K$). If the SU does not succeed in transmitting the K packets before the t_f time slots, then the whole frame is considered wasted. Since instantaneous channel gains and PU's activities are independent across time slots, the probability that the SU succeeds in transmitting the frame in t_f time slots or less is given by

$$P_{\text{frame}}\left(K, t_f\right) = \sum_{n=K}^{t_f} \binom{t_f}{n} p^n \left(1-p\right)^{t_f - n} \tag{2.15}$$

where p is the probability of transmitting a packet on some channel in a single time slot and is given by (2.3) or (2.21) if the SU's system was overlay or underlay respectively. $P_{\text{frame}}\left(K, t_f\right)$ represents the probability of finding K or more free time slots out of a total of t_f time slots.

In order to guarantee some QoS for the real-time data the SU needs to keep the probability of successful frame transmission above a minimum value denoted $r_{\min}$, that is $P_{\text{frame}} \geq r_{\min}$. Hence the problem becomes a throughput maximization problem subject to some average power and QoS constraints as follows

$$\begin{array}{ll} \text{maximize} & U_1(\mathbf{\Gamma}_{\text{th}}(1), \mathbf{P}_{1,1}) \\ \text{subject to} & S_1(\mathbf{\Gamma}_{\text{th}}(1), \mathbf{P}_{1,1}) \leq P_{\text{avg}} \\ & P_{\text{frame}}(K, t_f) \geq r_{\min} \\ \text{variables} & \mathbf{\Gamma}_{\text{th}}(1), \mathbf{P}_{1,1}. \end{array} \tag{2.16}$$

This is the optimization problem assuming an overlay system since we used equations (2.2) and (2.1) for the throughput and power, respectively. It can also be modified systematically to the case of an underlay system. Since there exists a one-to-one mapping between $P_{\text{frame}}(K, t_f)$ and p, then there exists a value for $\bar{D}_{\max}$ such that the inequality $p \geq 1/\bar{D}_{\max}$ is equivalent to the QoS inequality $P_{\text{frame}}\left(K, t_f\right) \geq r_{\min}$. That is, we can replace inequality $P_{\text{frame}}(K, t_f) \geq r_{\min}$ by $p \geq 1/\bar{D}_{\max}$ for some $\bar{D}_{\max}$ that depends on $r_{\min}$, K and t_f that are known a priori. Consequently, problem (2.16) is reduced to the simpler, yet equivalent, single-time-slot problem (2.4) and the SU can solve for $\mathbf{P}_1^*$ and $\mathbf{\Gamma}_{\text{th}}^*\left(1\right)$ vectors following the approach proposed in Section 2.2. The

SU solves this problem offline (i.e. before the beginning of the frame transmission) and uses this solution each time slot of the t_f time slots. With this offline scheme, the SU will be able to meet the QoS and the average power constraint requirements as well as maximizing its throughput.

2.3.2 Online Power-and-Threshold Adaptation

In problem (2.4), we have seen that as $1/\bar{D}_{\max}$ decreases, the system becomes less stringent in terms of the delay constraint. This results in an increase in the average throughput U_1^*. With this in mind, let us assume, in the generalized delay model, that at time slot 1 the SU succeeds in transmitting a packet. Thus, at time slot 2 the SU has $K-1$ remaining packets to be transmitted in $t_f - 1$ time slots. And from the properties of equation (2.15), $P_{\text{frame}}(K-1, t_f-1) > P_{\text{frame}}(K, t_f)$. This means that the system becomes less stringent in terms of the QoS constraint after a successful packet transmission. This advantage appears in the form of higher throughput. To see how we can make use of this advantage, define $P_{\text{frame}}(K(t), t_f - t + 1)$ as

$$P_{\text{frame}}\left(K(t), t_f - t + 1\right) = \\ \sum_{n=K(t)}^{t_f-t+1} \binom{t_f - t + 1}{n} \left(p(t)\right)^n \left(1 - p(t)\right)^{t_f - t + 1 - n}, \tag{2.17}$$

where $K(t)$ is the remaining number of packets before time slot $t \in \{1, ..., t_f\}$ and $p(t)$ is the probability of successful transmission at time slot t. At each time slot $t \in \{1, ...t_f\}$, the SU modifies the QoS constraint to be $P_{\text{frame}}(K(t), t_f - t + 1) \geq r_{\min}$ instead of $P_{\text{frame}}(K, t_f) \geq r_{\min}$, that was used in the offline adaptation, and solve the following problem

$$
\begin{aligned}
\text{maximize} \quad & U_1(\mathbf{\Gamma}_{\text{th}}(1), \mathbf{P}_{1,1}) \\
\text{subject to} \quad & S_1(\mathbf{\Gamma}_{\text{th}}(1), \mathbf{P}_{1,1}) \leq P_{\text{avg}} \\
& P_{\text{frame}}(K(t), t_f - t + 1) \geq r_{\min} \\
\text{variables} \quad & \mathbf{\Gamma}_{\text{th}}(1), \mathbf{P}_{1,1},
\end{aligned}
\tag{2.18}
$$

to obtain the power and threshold vectors. When the delay constraint in (2.18) is replaced by its equivalent constraint $p \geq 1/\bar{D}_{\max}$, the resulting problem can be solved using the overlay approach proposed in Section 2.2 without much increase in computational complexity since the power functions and thresholds are given in closed-form expressions. With this online adaptation, the average throughput U_1^* increases while still satisfying the QoS constraint.

2.4 Underlay System

In the overlay system, the SU tries to locate the free channels at each time slot to access these spectrum holes without interfering with the PUs. Recently, the FCC has allowed the SUs to interfere with the PU's network as long as this interference does not harm the PUs [50]. If the interference from the SU measured at the PU's receiver is below the tolerable level, then the interference is deemed acceptable.

In order to model the interference at the PR, we assume that the SU uses a channel sensing technique that produces the sufficient statistic z_i at channel i [51, 52]. The SU is assumed to know the distribution of z_i given channel i is free and busy, namely $f_{z|b}(z_i|b_i = 0)$ and $f_{z|b}(z_i|b_i = 1)$ respectively. For brevity, we omit the subscript i from b_i whenever it is clear from the context. The value of z_i indicates how confident the SU is in the presence of the PU at channel i. Thus the SU stops at channel i according to how likely busy it is and how much data rate it will gain from this channel (i.e. according to z_i and γ_i respectively). Hence, when the SU senses channel i to acquire z_i, the channel gain γ_i is probed and compared to some function $\gamma_{th}(i, z_i)$; if $\gamma_i \geq \gamma_{th}(i, z_i)$ transmission occurs on channel i, otherwise, channel i is skipped and $i+1$ is sensed. Potentially, $\gamma_{th}(i, z_i)$ is a function in the statistic z_i. This means that, at channel i, for each possible value that z_i might take, there is a corresponding threshold $\gamma_{\text{th}}(i, z)$. Before formulating the problem we note that this model can capture the overlay with sensing errors model as a special case where $f_{z|b}(z|b_i = 1) = (1 - P_{\text{MD}})\delta(z - z_{\text{b}}) + P_{\text{MD}}\delta(z - z_{\text{f}})$ while $f_{z|b}(z|b_i = 0) = P_{\text{FA}}\delta(z - z_{\text{b}}) + (1 - P_{\text{FA}})\delta(z - z_{\text{f}})$, where P_{MD} and P_{FA} are the probabilities of missed-detection and false-alarm respectively, while $\delta(z)$ is the Dirac delta function, and z_{b} and z_{f} that represent the values that z takes when the channel is busy and free, respectively. Hence, the interference constraint, which will soon be described, can be modified to a detection probability constraint and/or a false alarm probability constraint.

The SU's expected throughput is given by $U_1(\mathbf{\Gamma}_{\text{th}}(1, z), \mathbf{P}_1)$ which can be calculated recursively from

$$\begin{aligned} U_i \quad & (\mathbf{\Gamma}_{\text{th}}(i, z), \mathbf{P}_i) = \\ & c_i \int_{-\infty}^{\infty} \int_{\gamma_{\text{th}}(i,z)}^{\infty} \log(1 + P_i(\gamma)\gamma) f_\gamma(\gamma)\, d\gamma f_z(z)\, dz + \\ & p_i^{\text{skip}} U_{i+1}(\mathbf{\Gamma}_{\text{th}}(i+1, z), \mathbf{P}_{i+1}), \quad i \in \mathcal{M}, \end{aligned} \tag{2.19}$$

where $U_{M+1}(\mathbf{\Gamma}_{\text{th}}(M+1, z), \mathbf{P}_{M+1}) \triangleq 0$, $\mathbf{\Gamma}_{\text{th}}(i, z) \triangleq [\gamma_{\text{th}}(i, z), ..., \gamma_{\text{th}}(M, z)]^T$, $f_z(z) \triangleq \theta_i f_{z|b}(z|b_i = 0) + (1 - \theta_i) f_{z|b}(z|b_i = 1)$ is the PDF of the random variable z_i and $p_i^{\text{skip}} \triangleq \int_{-\infty}^{\infty} \int_0^{\gamma_{\text{th}}(i,z)} f_\gamma(\gamma)\, d\gamma f_z(z)\, dz$. The first term in (2.19) is the SU's throughput at channel i averaged over all realizations of z_i and that of $\gamma_i \geq \gamma_{\text{th}}(i, z)$. The second term is the average throughput when the SU skips channel i due to its low gain. Also, let the average interference from the SU's transmitter to the PU's receiver, aggregated over all M channels, be $I_1(\mathbf{\Gamma}_{\text{th}}(1, z), \mathbf{P}_1)$. This represents the total interference affecting the PU's network due to the existence of the SU. The SU is responsible for guaranteeing that this interference does not exceed a threshold I_{avg} dictated by the

PU's network. $I_1(\mathbf{\Gamma}_{\text{th}}(1,z),\mathbf{P}_1)$ can be derived using the following recursive formula

$$
\begin{aligned}
&I_i(\mathbf{\Gamma}_{\text{th}}(i,z),\mathbf{P}_i) = \\
&(1-\theta_i)\,c_i \int_{-\infty}^{\infty}\int_{\gamma_{\text{th}}(i,z)}^{\infty} P_i(\gamma)\, f_\gamma(\gamma)\, d\gamma f_{z|b}(z|b_i=1)\, dz \\
&+p_i^{\text{skip}} I_{i+1}(\mathbf{\Gamma}_{\text{th}}(i+1,z),\mathbf{P}_{i+1}), \qquad i\in\mathcal{M},
\end{aligned}
\tag{2.20}
$$

where $I_{M+1}(\mathbf{\Gamma}_{\text{th}}(M+1,z),\mathbf{P}_{M+1}) \triangleq 0$. This interference model is based on the assumption that the channel gain from the SU's transmitter to the PU's receiver is known at the SU's transmitter. This is the case for reciprocal channels when the PR acts as a transmitter and transmits training data to its intended primary transmitter (when it is acting as a receiver) [53]. The ST overhears this training data and estimates the channel from itself to the PR. Moreover, the gain at each channel from the ST to the PR is assumed unity for presentation simplicity. This could be extended easily to the case of non-unity-gain by multiplying the first term in (2.20) by the gain from the ST to the PR at channel i. Finally, $p_1(\mathbf{\Gamma}_{\text{th}}(1,z))$ is the probability of a successful transmission in the current time slot and can be calculated using

$$
\begin{aligned}
p_i(\mathbf{\Gamma}_{\text{th}}(i,z)) = \;& \int_{-\infty}^{\infty}\int_{\gamma_{\text{th}}(i,z)}^{\infty} f_\gamma(\gamma)\, d\gamma f_z(z)\, dz + \\
& p_i^{\text{skip}} p_{i+1}(\mathbf{\Gamma}_{\text{th}}(i+1,z)),
\end{aligned}
\tag{2.21}
$$

$i\in\mathcal{M}$, $p_{M+1}(\mathbf{\Gamma}_{\text{th}}(M+1,z)) \triangleq 0$. Given this background, the problem is

$$
\begin{aligned}
\text{maximize} \quad & U_1(\mathbf{\Gamma}_{\text{th}}(1,z),\mathbf{P}_1) \\
\text{subject to} \quad & I_1(\mathbf{\Gamma}_{\text{th}}(1,z),\mathbf{P}_1) \le I_{\text{avg}} \\
& p_1(\mathbf{\Gamma}_{\text{th}}(1,z)) \ge \tfrac{1}{\bar{D}_{\max}} \\
\text{variables} \quad & \mathbf{\Gamma}_{\text{th}}(1,z),\mathbf{P}_1,
\end{aligned}
\tag{2.22}
$$

Let λ_{I} and λ_{D} be the Lagrange multipliers associated with the interference and delay constraints of problem (2.22), respectively. Problem (2.22) is more challenging compared to the overlay case. This is because, unlike in (2.4), the thresholds in (2.22) are functions rather than constants. The KKT conditions for (2.22) are given by

$$P_i^*(\gamma) = \left(\frac{1}{\lambda_{\mathrm{I}}^* \mathrm{P}r\left[b_i = 1|z\right]} - \frac{1}{\gamma}\right)^+ \quad , \quad i \in \mathcal{M}. \tag{2.23}$$

$$\gamma_{\mathrm{th}}^*(i, z) = \frac{-\lambda_{\mathrm{I}}^* \mathrm{P}r\left[b_i = 1|z\right]}{W_0\left(-\exp\left(-\frac{\left(U_{i+1}^* - \lambda_{\mathrm{I}}^* I_{i+1}^* - \lambda_{\mathrm{D}}^*\left(1 - p_{i+1}^*\right)\right)^+}{c_i} - 1\right)\right)}, \quad i \in \mathcal{M}, \tag{2.24}$$

in addition to the primal feasibility, dual feasibility and the complementary slackness equations given in (2.8), (2.9) and (2.10), where $U_{i+1}^* \triangleq U_1\left(\mathbf{\Gamma}_{\mathrm{th}}^*(1, z), P_1^*(\gamma)\right)$, $I_{i+1}^* \triangleq I_1\left(\mathbf{\Gamma}_{\mathrm{th}}^*(1, z), P_1^*(\gamma)\right)$ and $p_{i+1}^* \triangleq p_1\left(\mathbf{\Gamma}_{\mathrm{th}}^*(1, z)\right)$ while $\mathrm{P}r\left[b_i = 1|z\right]$ is the conditional probability that channel i is busy given z_i and is given by

$$\mathrm{P}r\left[b_i = 1|z\right] = \frac{\left(1 - \theta_i\right) f_{z|b}\left(z|b_i = 1\right)}{f_z(z)}. \tag{2.25}$$

Note that $P_i^*(\gamma)$ is increasing in γ and is upper bounded by the term $1/\left(\lambda_{\mathrm{I}}^* \mathrm{P}r\left[b_i = 1|z\right]\right)$. Hence, as $\mathrm{P}r\left[b_i = 1|z\right]$ increases, the SU's maximum power becomes more limited, i.e. the maximum power decreases. This is because the PU is more likely to be occupying channel i. Thus the power transmitted from the SU should decrease in order to protect the PU.

Algorithm 1 can also be used to find λ_{I}^*. Only a single modification is required in the algorithm which is that S_1^* would be replaced by I_1^*. Thus the solution of problem (2.22) can be summarized on 3 steps: 1) Fix $\lambda_{\mathrm{D}}^* \in \mathbb{R}^+$ and find the corresponding optimum λ_{I}^* using the modified version of Algorithm 1. 2) Substitute the pair $\left(\lambda_{\mathrm{I}}^*, \lambda_{\mathrm{D}}^*\right)$ in equations (2.23) and (2.24) to get the power and threshold functions, then evaluate U_1^* from (2.19). 3) Repeat steps 1 and 2 for other values of λ_{D}^* until reaching the optimum λ_{D}^* that satisfies $p_1^* = 1/\bar{D}_{\max}$ and if there are multiple λ_{D}^*'s satisfying $p_1^* = 1/\bar{D}_{\max}$, then the optimum one is the one that gives the highest U_1^*. This approach yields the optimal solution. Next, Theorem 2 asserts the monotonicity of I_1^* in λ_{I}^* which allows using the bisection to find λ_{I}^* given some fixed λ_{D}^*.

Theorem 2. *I_1^* is decreasing in $\lambda_{\mathrm{I}}^* \in [0, \infty)$ given some fixed $\lambda_{\mathrm{D}}^* \geq 0$.*

Proof. We differentiate I_1^* with respect to λ_{I}^* given that $P_i^*(\gamma)$ and $\gamma_{\mathrm{th}}^*(i,z)$ are given by equations (2.23) and (2.24) respectively, then show that this derivative is negative. The proof is omitted since it follows the same lines of Theorem 1. □

Although the interference power constraint is sufficient for the problem to prevent the power functions from going to infinity, in some applications one may have an additional power constraint on the SUs. Hence, problem (2.22) can be modified to introduce an average power constraint that is given by $S_1(\mathbf{\Gamma}_{\mathrm{th}}(1,z), \mathbf{P}_1) \leq P_{\mathrm{avg}}$ where

$$\begin{aligned} S_i(\mathbf{\Gamma}_{\mathrm{th}}(i,z), \mathbf{P}_i) &= c_i \int_{-\infty}^{\infty} \int_{\gamma_{\mathrm{th}}(i,z)}^{\infty} P_i(\gamma)\, f_\gamma(\gamma)\, d\gamma f_z(z)\, dz \\ &\quad + p_i^{\mathrm{skip}} S_{i+1}(\mathbf{\Gamma}_{\mathrm{th}}(i+1,z), \mathbf{P}_{i+1}). \end{aligned} \tag{2.26}$$

It can be easily shown that the solution to the modified problem is similar to that presented in equations (2.23) and (2.24) which is

$$P_i^*(\gamma) = \left(\frac{1}{\lambda_{\mathrm{P}}^* + \lambda_{\mathrm{I}}^* \mathrm{P}r\left[b_i = 1 | z\right]} - \frac{1}{\gamma} \right)^+ , \tag{2.27}$$

$$\begin{aligned} &\gamma_{\mathrm{th}}^*(i,z) = \\ &\frac{-\left(\lambda_{\mathrm{P}}^* + \lambda_{\mathrm{I}}^* \mathrm{P}r\left[b_i = 1 | z\right]\right)}{W_0\left(-\exp\left(-\frac{\left(U_{i+1}^* - \lambda_{\mathrm{I}}^* I_{i+1}^* - \lambda_{\mathrm{P}}^* S_{i+1}^* - \lambda_{\mathrm{D}}^*\left(1 - p_{i+1}^*\right)\right)^+}{c_i} - 1\right)\right)}, \end{aligned} \tag{2.28}$$

$\forall i \in \mathcal{M}$ where $S_i^* \triangleq S_i(\mathbf{\Gamma}_{\mathrm{th}}^*(i,z), P_i^*(\gamma))$. This solution is more general since it takes into account both the average interference and the average power constraint besides the delay constraint. Moreover, it allows for the case where the power constraint is inactive which happens if the PU has a strict average interference constraint. In this case the optimum solution would result in $\lambda_{\mathrm{P}}^* = 0$ making equations (2.27) and (2.28) identical to equations (2.23) and (2.24) respectively.

2.5 Multiple Secondary Users

In this section, we show how our single SU framework can be extended to multiple SUs in a multiuser diversity framework without increase in the complexity of the algorithm. We will show that when the number of SUs is high, with slight modifications to the definitions of the throughput, power and probability of success, the single SU optimization problem in (2.4) (or (2.22)) can capture the multi-SU scenario. Moreover, the proposed solution for the overlay model still works for the multi-SU scenario. Finally, at the end of this section, we show that the proposed algorithm provides a throughput-optimal and delay-optimal solution with even a lower complexity for finding the thresholds compared to the single SU case, if the number of SUs is large.

Consider a CR network with L SUs associated with a centralized secondary base station (BS) in a downlink overlay scenario. Before describing the system model, we would like to note that when we say that channel i will be sensed, this means that each user will independently sense channel i and feedback the sensing outcome to the BS to make a global decision. Although we neglect sensing errors in this section, the analysis will work similarly in the presence of sensing errors by using the underlay model. At the beginning of each time slot the L SUs sense channel 1. If it is free, each SU observes it free with no errors and probes the instantaneous channel gain and feeds it back to the BS. The BS compares the maximum received channel gain among the L received channel gains to $\gamma_{\text{th}}(1)$. Channel 1 is assigned to the user having the maximum channel gain if this maximum gain is higher than $\gamma_{\text{th}}(1)$, while the remaining $L-1$ users continue to sense channel 2. On the other hand if the maximum channel gain is less than $\gamma_{\text{th}}(1)$, channel 1 is skipped and channel 2 is sensed by all L users. Unlike the case in the single SU scenario where only a single channel is claimed per time slot, in this multi-SU system, the BS can allocate more than one channel in one time slot such that each SU is not allocated more than one channel and each channel is not allocated to more than one SU. Based on this scheme, the expected per-SU throughput U_1^L is calculated from

$$\begin{aligned} U_i^l = & \frac{\theta_i c_i}{l} \int_{\gamma_{\text{th}}(i)}^{\infty} \log\left(1 + P_{1,i}(\gamma)\gamma\right) f_l(\gamma)\, d\gamma + \\ & \theta_i \bar{F}_l(\gamma_{\text{th}}(i)) \left(1 - \frac{1}{l}\right) U_{i+1}^{l-1} + \left(1 - \theta_i \bar{F}_l(\gamma_{\text{th}}(i))\right) U_{i+1}^l \end{aligned} \tag{2.29}$$

$i \in \mathcal{M}$ and $l \in \{L - i + 1, ..., L\}$ with initialization $U_{M+1}^l = 0$. Here $f_l(\gamma)$ represents the density of the maximum gain among l i.i.d. users' gains, while $\bar{F}_l(\gamma)$ is its complementary cumulative distribution function. We study the case where $L \gg M$, thus when a channel is allocated to a user we can assume that the remaining number of users is still L. Thus we approximate l with L $\forall l \in \{L - i, ..., L\}$ and $\forall i \in \mathcal{M}$. Similar to the the throughput derived in (2.29), we could write the exact expressions for the per-SU average power and per-SU probability of transmission. And since $L \gg M$, we can approximate S_i^l with S_i^L and p_i^l with p_i^L, $\forall l \in \{L - i + 1, ..., L\}$ and $\forall i \in \mathcal{M}$. The per-SU expected throughput U_1^L, the average power S_1^L and the probability of transmission p_1^L can be derived from

$$\begin{aligned} U_i^L(\mathbf{\Gamma}_{\text{th}}(i), \mathbf{P}_{1,i}) = & \frac{\theta_i c_i}{L} \int_{\gamma_{\text{th}}(i)}^{\infty} \log\left(1 + P_{1,i}(\gamma)\gamma\right) f_L(\gamma)\, d\gamma + \\ & \left[1 - \frac{\theta_i \bar{F}_L(\gamma_{\text{th}}(i))}{L}\right] U_{i+1}^L\left(\mathbf{\Gamma}_{\text{th}}(i+1), \mathbf{P}_{i+1}\right) \end{aligned} \tag{2.30}$$

$$\begin{aligned} S_i^L(\mathbf{\Gamma}_{\text{th}}(i), \mathbf{P}_{1,i}) = & \frac{\theta_i c_i}{L} \int_{\gamma_{\text{th}}(i)}^{\infty} P_{1,i}(\gamma) f_L(\gamma)\, d\gamma + \\ & \left[1 - \frac{\theta_i \bar{F}_L(\gamma_{\text{th}}(i))}{L}\right] S_{i+1}^L(\mathbf{\Gamma}_{\text{th}}(i+1), \mathbf{P}_{i+1}), \end{aligned} \tag{2.31}$$

$$\begin{aligned} p_i^L(\mathbf{\Gamma}_{\text{th}}(i)) = & \frac{\theta_i}{L} \bar{F}_L(\gamma_{\text{th}}(i)) + \\ & \left[1 - \frac{\theta_i \bar{F}_L(\gamma_{\text{th}}(i))}{L}\right] p_{i+1}^L(\mathbf{\Gamma}_{\text{th}}(i+1)), \end{aligned} \tag{2.32}$$

$i \in \mathcal{M}$, respectively, with $U_{M+1}^{L} = S_{M+1}^{L} = p_{M+1}^{L} = 0$. To formulate the multi-SU optimization problem, we replace U_1, S_1 and p_1 in (2.4) with U_1^L, S_1^L and p_1^L derived in equations (2.30), (2.31) and (2.32), respectively. Taking the Lagrangian and following the same procedure as in Section 2.2, we reach at the solution for $P_{1,i}^{*}$ and $\gamma_{\text{th}}^{*}(i)$ as given by equations (2.6) and (2.12) respectively. Hence, equations (2.6) and (2.12) represent the optimal solution for the multi-SU scenario. The details are omitted since they follow those of the single SU case discussed in Section 2.2.

Next we show that this solution is optimal with respect to the delay as well as the throughput when L is large. We show this by studying the system after ignoring the delay constraint and show that the resulting solution of this system (which is what we refer to as the unconstrained problem) is a delay optimal one as well. The solution of the unconstrained problem is given by setting $\lambda_{\text{D}}^{*} = 0$ in (2.12) arriving at

$$\gamma_{\text{th}}^{*}(i)\,|_{\lambda_{\text{D}}^{*}=0} = \frac{-\lambda_{\text{P}}^{*}}{W_0\left(-\exp\left(-\frac{\left(U_{i+1}^{L*}-\lambda_{\text{P}}^{*}S_{i+1}^{L*}\right)^{+}}{c_i}-1\right)\right)}. \tag{2.33}$$

$\forall i \in \mathcal{M}$. As the number of SUs increases, the per-user expected throughput U_1^L decreases since these users share the total throughput. Moreover, U_i^L decreases as well $\forall i \in \mathcal{M}$ decreasing the value of $\gamma_{\text{th}}^{*}(i)$ (from equation (2.33) until reaching its minimum (i.e. $\gamma_{\text{th}}^{*}(i) = \lambda_{\text{P}}^{*}$) (the right-hand-side of (2.33) is minimum when its denominator is as much negative as possible. That is, when $W_0(x) = -1$ since $W_0(x) \geq -1, \forall x \in \mathbb{R}$) as $L \to \infty$. From (2.32), it can be easily shown that $p_1^L(\mathbf{\Gamma}_{\text{th}}(1))$ is monotonically decreasing in $\gamma_{\text{th}}(i)$ $\forall i \in \mathcal{M}$. Thus the minimum possible average delay (corresponding to the maximum $p_1^L(\mathbf{\Gamma}_{\text{th}}(1))$) occurs when $\gamma_{\text{th}}(i)$ is at its minimum possible value for all $i \in \mathcal{M}$. Consequently, having $\gamma_{\text{th}}^{*}(i) = \lambda_{\text{P}}^{*}$ means that the system is at the optimum delay point. That is, the unconstrained problem cannot achieve any smaller delay with an additional delay constraint. Hence, the multi-

SU problem, that is formulated by adding a delay constraint to the unconstrained problem, achieves the optimum delay performance when L is asymptotically large.

Recall that the overall complexity of solution for the single SU case is due to three factors: 1) evaluating the Lambert W function in Algorithm 1, 2) the bisection algorithm in Algorithm 1 and 3) the search over λ_{D}. On the other hand, the complexity of solution for the multi-SU case decreases asymptotically (as $L \to \infty$). This is because of two reasons: 1) When $L \gg M$, $\gamma_{\mathrm{th}}^{*}(i) \to \lambda_{\mathrm{P}}^{*} \forall i \in \mathcal{M}$. Which means that we will not have to evaluate the Lambert W function in (2.12) but instead we set $\gamma_{\mathrm{th}}^{*}(i) = \lambda_{\mathrm{P}}^{*}$, since $L \gg M$. 2) When $\gamma_{\mathrm{th}}^{*}(i) = \lambda_{\mathrm{P}}^{*}$ there will be no need to find λ_{D}^{*} since the delay is minimum (we recall that in the single SU case, we need to calculate λ_{D}^{*} to substitute it in (2.12) to evaluate $\gamma_{\mathrm{th}}^{*}(i)$, but in the multi-SU case $\gamma_{\mathrm{th}}^{*}(i) = \lambda_{\mathrm{P}}^{*}$).

2.6 Numerical Results

We show the performance of the proposed solution for the overlay and underlay scenarios. The slot duration is taken to be unity (i.e. all time measurements are taken relative to the time slot duration), while $\tau = 0.05T$. Here, we use $M = 10$ channels that suffer i.i.d. Rayleigh fading. The availability probability is taken as $\theta_i = 0.05i$ throughout the simulations. The power gain γ is exponentially distributed as $f_\gamma(\gamma) = \exp(\gamma/\bar{\gamma})/\bar{\gamma}$ where $\bar{\gamma}$ is the average channel gain and is set to be 1 unless otherwise specified.

Fig. 2.2 plots the expected throughput U_1^* for the overlay scenario after solving problem (2.4). U_1^* is plotted using equation (2.2) that represents the average number of bits transmitted divided by the average time required to transmit those bits, taking into account the time wasted due to the blocked time slots. We plot U_1^* with $\bar{D}_{\max} = 1.02T$ and with $\bar{D}_{\max} = \infty$ (i.e. neglecting the delay constraint). We refer to the former problem as constrained problem, while to the latter as unconstrained problem.

We also compare the performance to the non optimum stopping rule case (No-OSR) where the SU transmits at the first available channel. We expect the No-OSR case to have the best delay and the worst throughput performances. We can see that the unconstrained problem has the best throughput amongst all constrained problems.

Examining the constrained problem for different sensing orders of the channels, we observe that when the channels are sorted in an ascending order of θ_i, the throughput is higher. This is because a channel i has a higher chance of being skipped if put at the beginning of the order compared to the case if put at the end of the order. This is a property of the problem no matter how the channels are ordered, i.e. this property holds even if all channels have equal values of θ_i. Hence, it is more favorable to put the high quality channels at the end of the sensing order so that they are not put in a position of being frequently skipped. However, this is not necessarily optimum order, which is out of the scope of this work and is left as a future work for this delay-constrained optimization problem.

We also plot the expected throughput of a simple stopping rule that we call K-out-of-M scheme, where we choose the highest K channels in availability probability and ignore the remaining channels as if they do not exist in the system. The SU senses those K channels sequentially, probes the gain of each free channel, if any, and transmits on the channel with the highest gain. This scheme has a constant fraction $K\tau/T$ of time wasted each slot. Yet it has the advantage of choosing the best channel among multiple available ones. In Fig. 2.2 we can see that the degradation of the throughput when $K = 5$ compared to the optimal stopping rule scheme. The reason is two-fold: 1) Due to the constant wasted fraction of time, and 2) Ignoring the remaining channels that could potentially be free with a high gain if they were considered as opposed to the constrained problem.

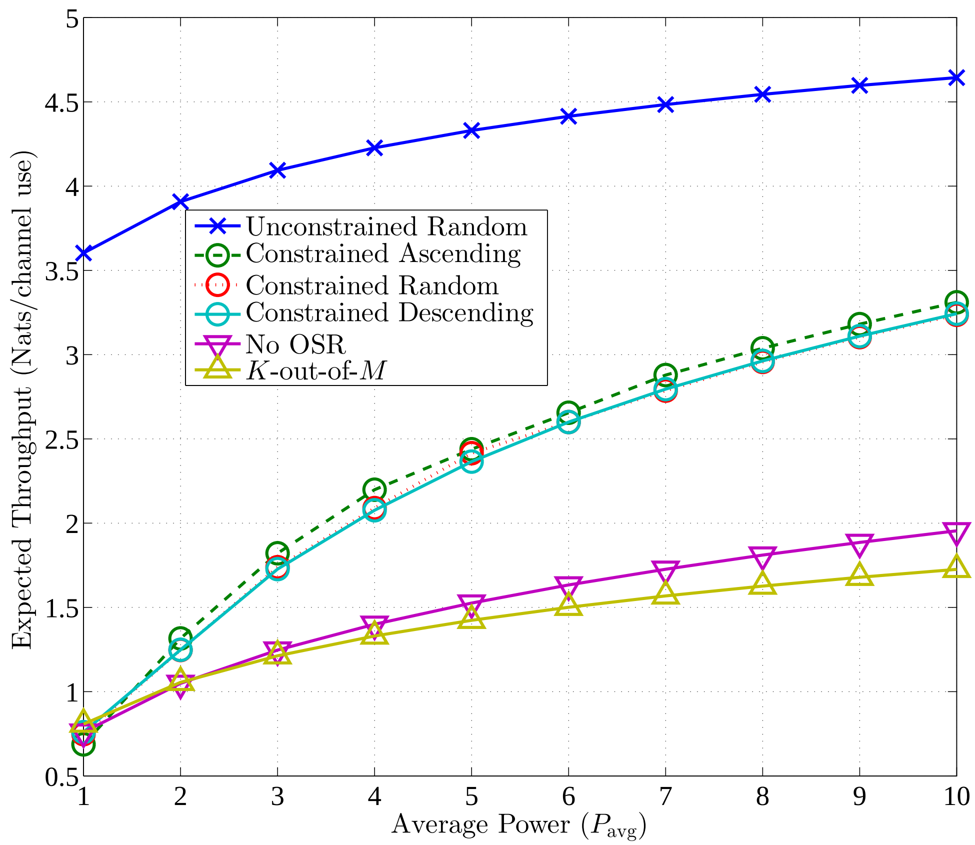

Figure 2.2: The expected throughput for the overlay scenario for four cases: 1) Proposed constrained problem: with average delay constraint for three channel ordering possibilities (ascending ordering of channel availability probabilities, descending ordering, and random ordering), 2) Unconstrained problem that ignores the delay constraint, 3) No optimum stopping rule (No-OSR) where the SU transmits at the first free channel and 4) K-out-of-M scheme where the SU assumes the system has only $K = 5$ channels and ignores the remaining $M - K$ channels.

The delay is shown in Fig. 2.3 for both the constrained and the unconstrained problems. We see that the unconstrained problem suffers around 6% increase in the delay, at $P_{\text{avg}} = 10$, compared to the constrained one.

Studying the system performance under low average channel gain is essential. A low average channel gain represents a SU's channel being in a permanent deep fade or if there is a relatively high interference level at the secondary receiver. Fig.

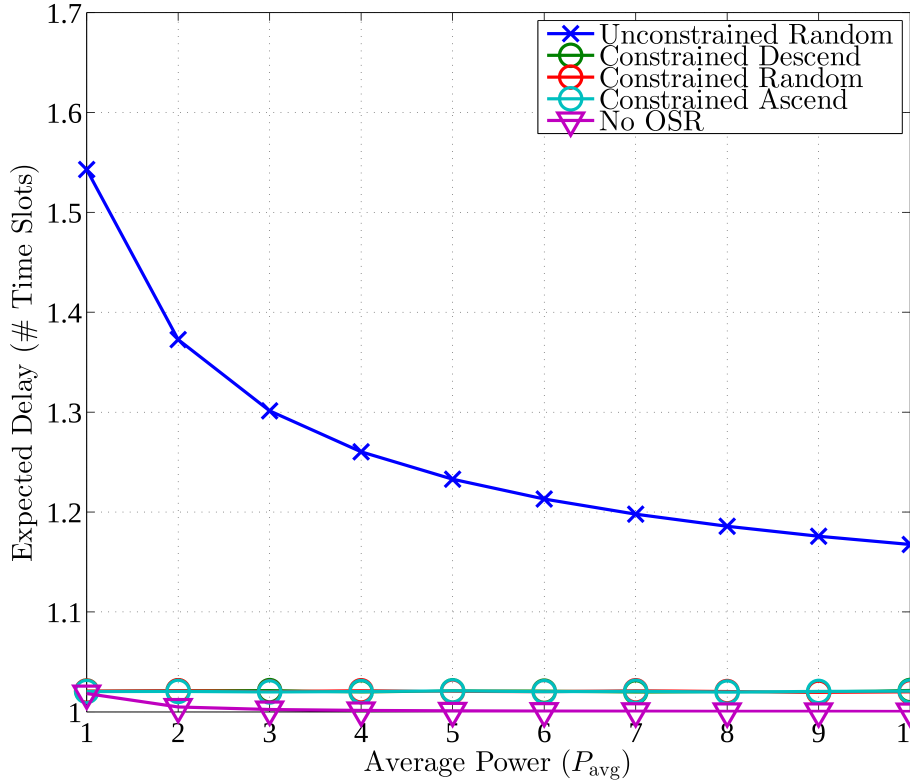

Figure 2.3: The expected delay for the overlay scenario for problem (2.4). The unconstrained problem can suffer arbitrary high delay. The constrained problem has a guaranteed average delay for all ordering strategies. The No-OSR scenario, on the other hand, has the best delay performance since the SU uses the first free channel.

2.4 shows $\gamma^*_{\mathrm{th}}(i)$ versus the $\bar{\gamma}$. At low $\bar{\gamma}$, the throughput is expected to be small, hence $\gamma^*_{\mathrm{th}}(i)$ is close to its minimum value λ^*_{P} so that even if γ_i is relatively small, i should not be skipped. In other words, at low average channel gain, the expected throughput is small, thus a relatively low instantaneous gain will be satisfactory for stopping at channel i. While when the average channel gain increases, $\gamma^*_{\mathrm{th}}(i)$ should increase so that only high instantaneous gains should lead to stopping at channel i. In both cases, high and low $\bar{\gamma}$ there still is a trade-off between choosing a high versus a low value of $\gamma^*_{\mathrm{th}}(i)$.

The sensing channel (i.e. the channel between the PT and ST over which

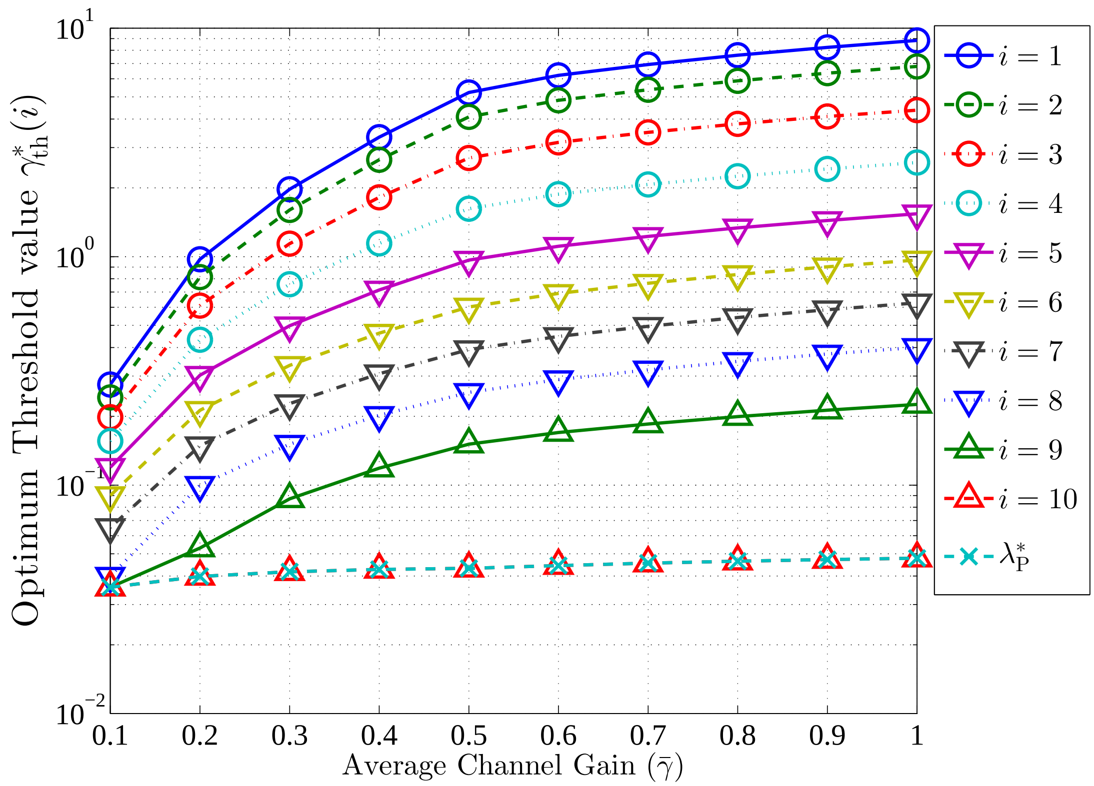

Figure 2.4: The gap between the optimum threshold $\gamma^*_{\mathrm{th}}(i)$ and its minimum value λ^*_{P} increases as the average gain increases. This is because as $\bar{\gamma}$ increases, U_{i+1} increases as well. Hence $\gamma^*_{\mathrm{th}}(i)$ increases so that only sufficiently high instantaneous gains should lead to stopping at channel i.

the ST overhears the PT activity) is modeled as AWGN with unit variance. The distributions of the energy detector output z (average energy of N samples sampled from this sensing channel) under the free and busy hypotheses are the Chi-square and a Noncentral Chi-square distributions given by

$$f_{z|b}\left(z|b_i=0\right) = \left(\frac{N}{\sigma^2}\right)^N \frac{z^{N-1}}{(N-1)!} \exp\left(\frac{-Nz}{\sigma^2}\right), \tag{2.34}$$

$$f_{z|b}\left(z|b_i=1\right) =$$
$$\left(\frac{N}{\sigma^2}\right)\left(\frac{z}{\mathcal{E}}\right)^{\frac{N-1}{2}} \exp\left(\frac{-N\left(z+\mathcal{E}\right)}{\sigma^2}\right) I^{\mathrm{Bes}}_{N-1}\left(\frac{2N\sqrt{\mathcal{E}z}}{\sigma^2}\right), \tag{2.35}$$

where σ^2, which is set to 1, is the variance of the Gaussian noise of the energy detector,

$\mathcal{E}$ is the amount of energy received by the ST due to the activity of the PT and is taken as $\mathcal{E} = 2\sigma^2$ throughout the simulations, while $I_i^{\text{Bes}}(x)$ is the modified Bessel function of the first kind and ith order, and $N = 10$.

The main problem we are addressing in this chapter is the optimal stopping rule that dictates for the SU when to stop sensing and start transmitting. As we have seen, this is identified by the threshold vector $\mathbf{\Gamma}^*_{\text{th}}(1, z)$. If the SU does not consider the optimal stopping rule problem and rather transmits as soon as it detects a free channel, then it will be wasting future opportunities of possibly higher throughput. Hence, we expect a degradation in the throughput. We plot the two scenarios in Fig. 2.5 for the underlay system with no delay constraint.

Throughout this chapter, we use bold fonts for vectors and asterisk to denote that x^* is the optimal value of x; all logarithms are natural, while the expected value operator is denoted $\mathbb{E}[\cdot]$ and is taken with respect to all the random variables in its argument. Finally, we use $(x)^+ \triangleq \max(x, 0)$ and $\mathbb{R}$ to denote the set of the real numbers.

For the multiple SU scenario, the numerical analysis were run for the case of $L = 30$ SUs while $M = 10$ channels. We assumed the fading channels are i.i.d. among users and among frequency channels. Each channel is exponentially distributed with unity average channel gain. And since L is large, the distribution of the maximum gain among L random gains converges in distribution to the Gumbel distribution [54] having a cumulative distribution function of $\exp(-\exp(-\gamma/\bar{\gamma}))$. The per-user throughput U_1^{L*} is plotted in Fig. 2.6 where the throughput of the delay-constrained and of the unconstrained optimization problems coincide. This is because when $L \gg M$, the solution of the unconstrained problem is delay optimal as well. Hence, adding a delay constraint does not sacrifice the throughput, when L is large. Moreover, the

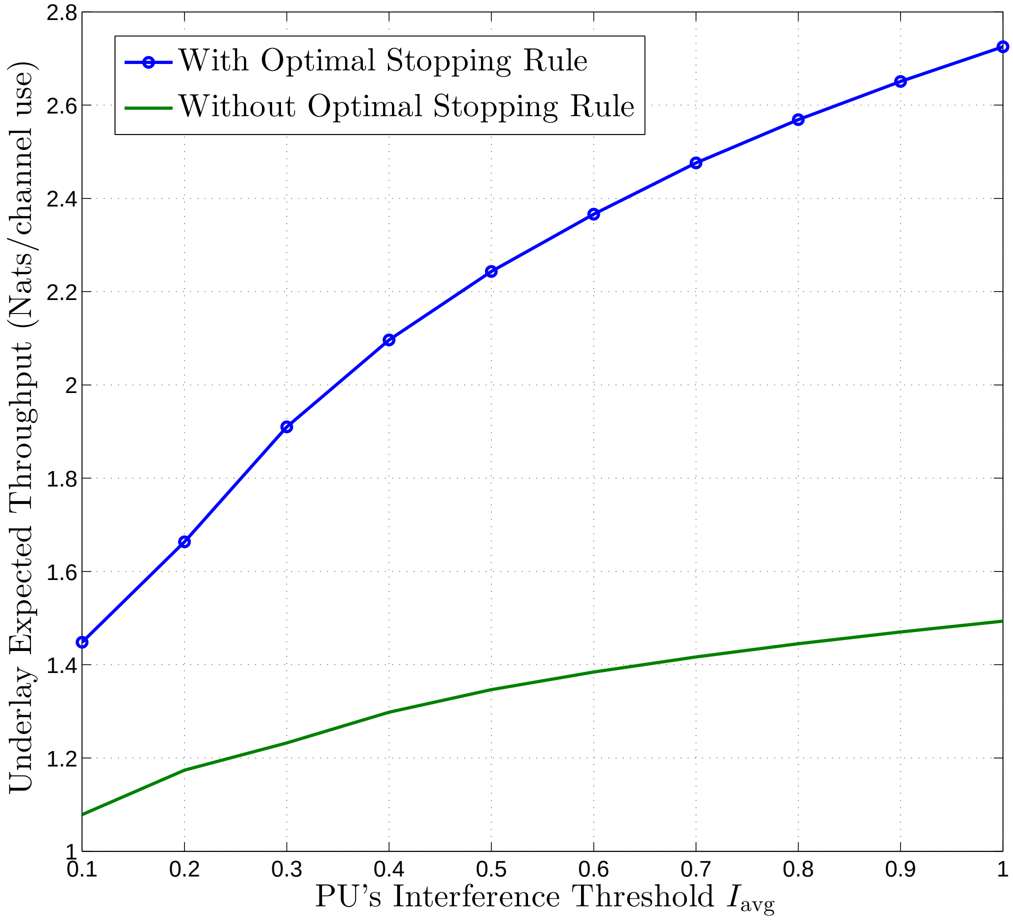

Figure 2.5: The underlay expected throughput versus the average interference threshold I_{avg}. Two scenarios are shown: with and without the optimal stopping rule formulation. In the latter, the SU transmits as soon as a channel is found free.

delay performance shown in Fig. 2.7 shows that the delay does not change with and without considering the average delay constraint since the system is delay- (and throughput-) optimal already.

We have simulated the system for the online algorithm of Section 2.3 for $K(1) = 2$ packets and $t_f = 4$ time slots. We simulated the system at $r_{\min} = 0.95$ which means that the QoS of the SU requires that at least 95% of the frames to be successfully transmitted. Fig. 2.8 shows the improvement in the throughput of

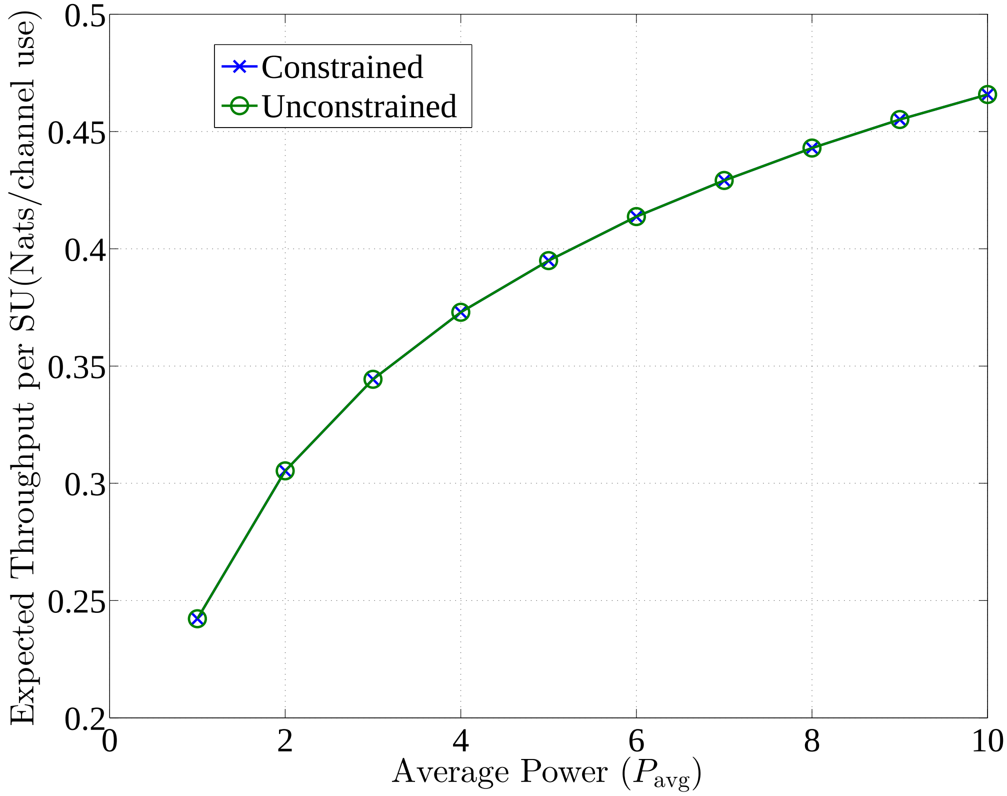

Figure 2.6: Per user throughput of the system at $L = 30$ SUs. The throughput of the constrained and unconstrained problem coincide since the system is throughput (and delay) optimal.

the online over the offline adaptation. This is because the SU adapts the power and thresholds at each time slot to allocate the remaining resources (i.e. remaining time slots) according to the remaining number of packets and the desired QoS. This comes at the expense of re-solving the problem at each time slot (i.e. t_f times more).

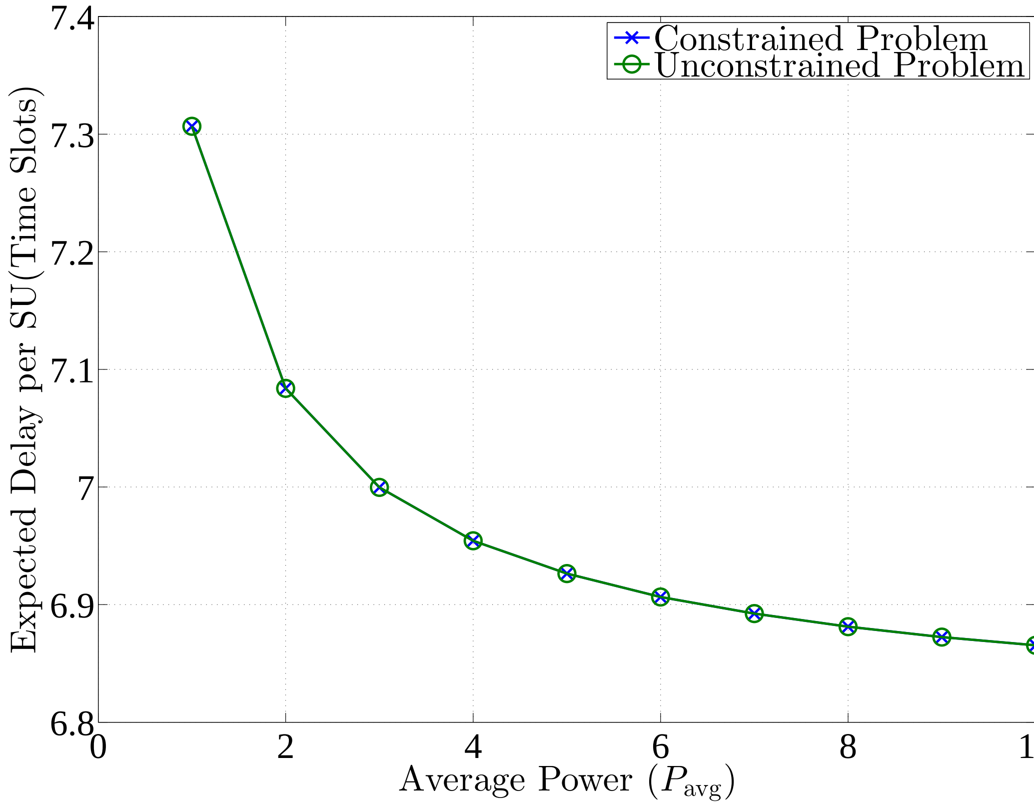

Figure 2.7: The average delay seen by each user in the system at $L = 30$ SUs. The delay of the constrained and unconstrained problems coincide since the system is delay (and throughput) optimal.

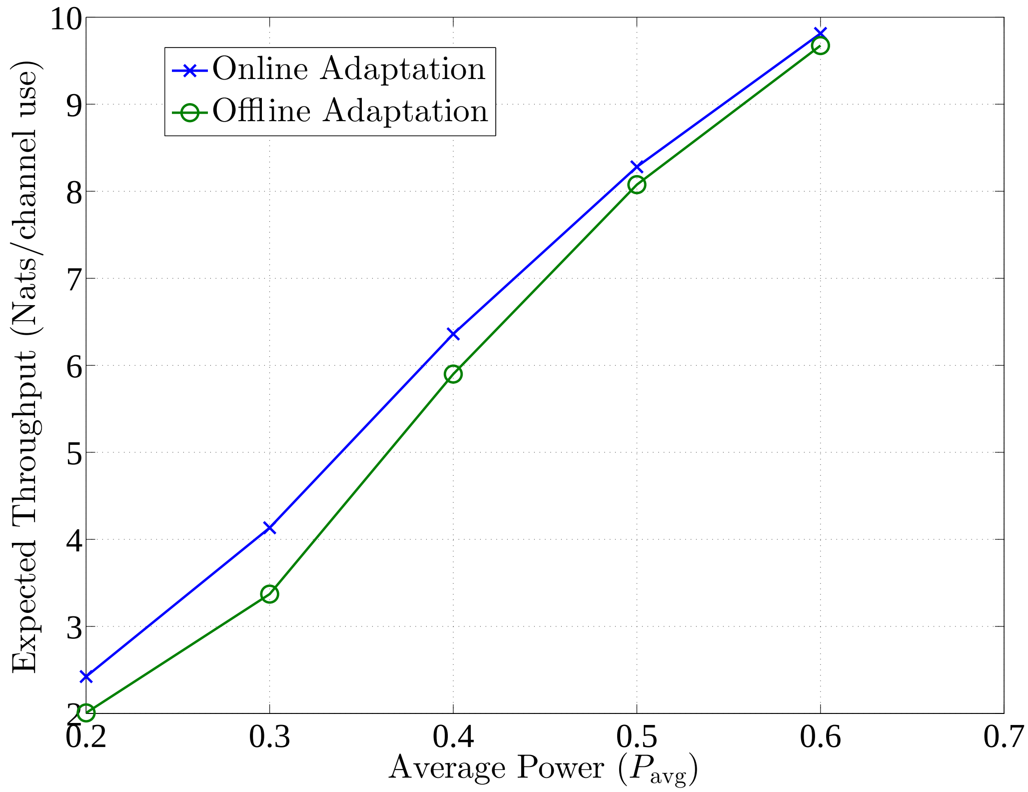

Figure 2.8: The performance of the online adaptation algorithm for the general delay model.

Chapter 3

AVERAGE-DELAY FRAMEWORK

In this chapter we study the delay resulting from the service time and the queue-waiting time. The service time is affected by the power transmitted by the SU, while the queue-waiting time is affected by the transmitted power as well as the scheduling algorithm. We propose a delay-optimal scheduling-and-power-allocation algorithm that guarantees bounds on the SUs' delays while causing an acceptable interference to the PUs. This algorithm is useful to provide fair delay guarantees to the SUs when delay fairness cannot be achieved due to the heterogeneity in SUs' channel statistics. The contributions in this chapter are: i) Proposing a joint power-control and scheduling algorithm that is optimal with respect to the average delay of the SUs in an interference-limited system; ii) Showing that the proposed algorithm can provide differentiated service to the different SUs based on their heterogeneous QoS requirements. Moreover, the complexity of the algorithm is shown to be polynomial in time since it is equivalent to that of sorting a vector of N numbers, where N is the number of SUs in the system.

3.1 Network Model

We assume a CR system consisting of a single secondary base station (BS) serving N secondary users (SUs) indexed by the set $\mathcal{N} \triangleq \{1, \cdots N\}$ (Fig. 3.1). We are considering the uplink phase where each SU has its own queue buffer for packets that need to be sent to the BS. The SUs share a single frequency channel with a single PU that has licensed access to this channel. The CR system operates in an underlay fashion where the PU is using the channel continuously at all times. SUs are allowed to transmit as long as they do not cause harmful interference to the PU. In this work, we consider two different scenarios where the interference can be considered as

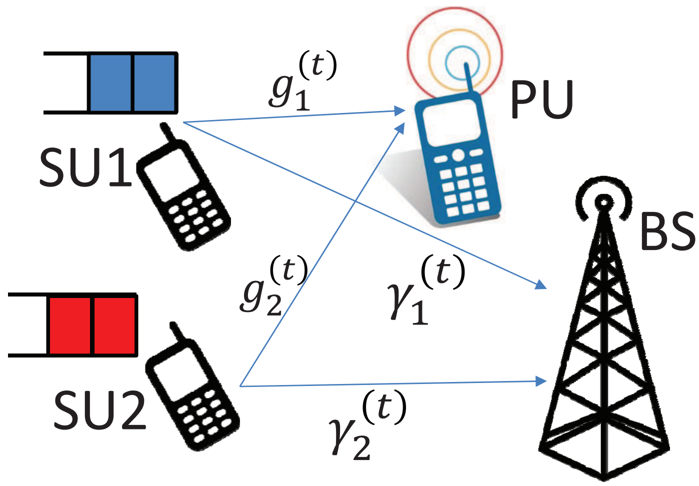

Figure 3.1: The CR system considered is an uplink one with N SUs (in this figure $N = 2$) communicating with their BS. There exists an interference link between each SU and the existing PU. The PU is assumed to be using the channel continuously.

harmful. The first is an instantaneous interference constraint where the interference received by the PU at any given slot should not exceed a prespecified threshold I_{inst}, while the second is an average interference constraint where the interference received by the PU averaged over a large duration of time should not exceed a prespecified threshold I_{avg}. Moreover, in order for the secondary BS to be able to decode the received signal, no more than one SU at a time slot is to be assigned the channel for transmission.

3.1.1 Channel and Interference Model

We assume a time slotted structure where each slot is of duration T seconds, and equal to the coherence time of the channel. The channel between SU i and the BS is block fading with instantaneous power gain $\gamma_i^{(t)}$, at time slot t, following the probability mass function $f_{\gamma_i}(\gamma)$ with mean $\bar{\gamma}_i$ and i.i.d. across time slots, and $\gamma_{\max}$ is the maximum gain $\gamma_i^{(t)}$ could take. The channel gain is also independent across SUs but not necessarily identically distributed allowing heterogeneity among users. SUs use a rate adaptation scheme based on the channel gain $\gamma_i^{(t)}$. The transmission rate of SU i at time slot t is

$$R_i^{(t)} = \log\left(1 + P_i^{(t)}\gamma_i^{(t)}\right) \quad \text{bits}, \tag{3.1}$$

where $P_i^{(t)}$ is the power by which SU i transmits its bits at slot t. We assume that there exists a finite maximum rate $R_{\max}$ that the SU cannot exceed. This rate is dictated by the maximum power $P_{\max}$ and the maximum channel gain $\gamma_{\max}$.

The PU experiences interference from the SUs through the channel between each SU and the PU. The interference channel between SU i and the PU, at slot t, has a power gain $g_i^{(t)}$ following the probability mass function $f_{g_i}(g)$ with mean $\bar{g}_i$, and having $g_{\max}$ as the maximum value that $g_i^{(t)}$ could take. These power gains are assumed to be independent among SUs but not identically distributed. We assume that SU i knows the value of $\gamma_i^{(t)}$ as well as $g_i^{(t)}$, at the beginning of slot t through some channel estimation phase [55]. The channel estimation to acquire $g_i^{(t)}$ can be done by overhearing the pilots transmitted by the primary receiver when it is acting as a transmitter [55, Section VI]. The channel estimation phase is out of the scope of this work, however the effect of channel estimation errors will be discussed in Section 4.6.

3.1.2 Queuing Model

3.1.2.1 Arrival Process

We assume that packets arrive to the SU i's buffer at the beginning of each slot. The number of packets arriving to SU i's buffer follows a Bernoulli process with a fixed parameter λ_i packets per time slot. Following the literature, packets are buffered in infinite-sized buffers [56, pp.163] and are served according to the first-come-first-serve discipline. Each packet has a fixed length of L bits that is constant for all users. In this chapter, we study the case where $L \gg R_{\max}$ which is a typical case for packets with large sizes as video packets [57, Section 3.1.6.1]. Due to the randomness in the channels, each packet takes a random number of time slots to be completely transmitted to the BS. This depends on the rate of transmission $R_i^{(t)}$ as will be explained next.

3.1.2.2 Service Process

When SU i is scheduled for transmission at slot t, it transmits $M_i^{(t)}$ bits of the head-of-line (HOL) packet of its queue. The remaining bits of this HOL packet remain in the HOL of SU i's queue until it is reassigned the channel in subsequent time slots. $M_i^{(t)}$ is given by

$$M_i^{(t)} \triangleq \min\left(R_i^{(t)}, L_i^{\text{rem}}(t)\right) \quad \text{bits, and} \tag{3.2}$$

respectively, where $L_i^{\text{rem}}(t)$ is the remaining number of bits of the HOL packet at SU i at the beginning of slot t. $L_i^{\text{rem}}(t)$ is initialized by L whenever a packet joins the HOL position of SU i's queue so that it always satisfies $0 \leq L_i^{\text{rem}}(t) \leq L$, $\forall t$. A packet is not considered transmitted unless all its L bits are transmitted, at which point SU i's queue decreases by 1 packet. At the beginning of slot $t+1$ the following packet in the buffer, if any, becomes SU i's HOL packet and $L_i^{\text{rem}}(t+1)$ is reset back to L bits. $L_i^{\text{rem}}(t)$ is given by

$$L_i^{\text{rem}}(t+1) \triangleq \begin{cases} L\mathbb{1}\left(Q_i(t) + |\mathcal{A}_i^{(t+1)}| > 0\right) & \text{if } L_i^{\text{rem}}(t) = M_i^{(t)} \\ L_i^{\text{rem}}(t) - M_i^{(t)} & \text{otherwise} \end{cases} \tag{3.3}$$

where $\mathcal{A}_i^{(t)}$ is the set carrying the index of the packet, if any, arriving to SU i at the beginning of slot t, thus $|\mathcal{A}_i^{(t)}|$ is either 0 or 1 since at most one packet per slot can arrive to SU i, $\mathbb{1}(x)$ is the indicator function which is 1 if the event x occurs and 0 otherwise, while $Q_i(t)$ represents the number of packets in SU i's queue at the beginning of slot t that evolves as follows

$$Q_i(t+1) = \left(Q_i(t) + |\mathcal{A}_i^{(t)}| - S_i^{(t)}\right)^+, \tag{3.4}$$

where the packet service indicator $S_i^{(t)} = 1$ if $L_i^{\text{rem}}(t) = M_i^{(t)}$.

The service time s_i of SU i is the number of time slots required to transmit one packet for SU i, excluding the service interruptions. It can be shown that the average service time for user i is $L/\,\mathbb{E}\left[R_i^{(t)}\right]$ time slots per packet where the expectation is taken over the channel gain $\gamma_i^{(t)}$ as well as over the power $P_i^{(t)}$ when it is channel dependent and random. One example of a random power policy is the *channel inversion* policy as will be discussed later (see equation (3.15)). The service time is assumed to follow a general distribution, throughout the chapter, that depends on the distribution of $P_i^{(t)}\gamma_i^{(t)}$.

We define the delay $W_i^{(j)}$ of a packet j as the total amount of time, in time slots, packet j spends in SU i's buffer from the slot it joined the queue until the slot when its last bit is transmitted. The time-average delay experienced by SU i's packets is given by [21]

$$\bar{W}_i \triangleq \lim_{T\to\infty} \frac{\mathbb{E}\left[\sum_{t=1}^{T} \sum_{j\in\mathcal{A}_i^{(t)}} W_i^{(j)}\right]}{\mathbb{E}\left[\sum_{t=1}^{T} |\mathcal{A}_i^{(t)}|\right]} \tag{3.5}$$

which is the expected total amount of time spent by all packets arriving in a time interval, of a large duration, normalized by the expected number of packets that arrived in this interval.

3.1.3 Transmission Process

At the beginning of each time slot t, the BS schedules a SU and broadcasts its index i^* and its power $P_{i^*}^{(t)}$ to all SUs on a common control channel. SU i^*, in turn, begins transmission of $M_{i^*}^{(t)}$ bits of its HOL packet with a constant power $P_{i^*}^{(t)}$. We assume the BS receives these bits error-free by the end of slot t then a new time slot $t+1$ starts. In this chapter, our main goal is the selection of the SU i^* which is a scheduling problem, as well as the choice of the power $P_{i^*}^{(t)}$ which is power allocation. We now elaborate further on this problem.

3.2 Problem Statement

Each SU i has an average delay constraint $\bar{W}_i \leq d_i$ that needs to be satisfied. Moreover, there is an interference constraint that the SU needs to meet in order to coexist with the PU. We discuss the two different constraints and state the problem associated with each constraint.

3.2.1 Instantaneous Interference Constraint

Under the instantaneous interference constraint, the main objective is to solve the following problem

$$\begin{aligned}
\underset{\{i^{*(t)}\},\{\mathbf{P}^{(t)}\}}{\text{minimize}} \quad & \textstyle\sum_{i=1}^{N} \bar{W}_i \\
\text{subject to} \quad & \textstyle\sum_{i=1}^{N} P_i^{(t)} g_i^{(t)} \leq I_{\text{inst}} \quad , \quad \forall t \geq 1 \\
& \bar{W}_i \leq d_i \\
& P_i^{(t)} \leq P_{\max} \ , \ \forall i \in \mathcal{N} \ \text{ and } \ \forall t \geq 1, \\
& \textstyle\sum_{i=1}^{N} \mathbb{1}\left(P_i^{(t)}\right) \leq 1 \quad , \quad \forall t \geq 1,
\end{aligned} \tag{3.6}$$

where $\mathbf{P}^{(t)} \triangleq [P_1^{(t)}, \cdots, P_N^{(t)}]^T$, $\{i^{*(t)}\}$ represents the scheduler at each time slot $t \geq 1$, while $\mathbb{1}(x) \triangleq 1$ if $x \neq 0$ and 0 otherwise. The last constraint indicates that no more than a single SU is to be transmitting at slot t.

3.2.2 Average and Instantaneous Interference Constraint

Let I denote the long-term average interference received by the PU given by

$$I \triangleq \lim_{T\to\infty} \sum_{i=1}^{N} \frac{1}{T} \sum_{t=1}^{T} P_i^{(t)} g_i^{(t)}. \tag{3.7}$$

The following problem is the same as (3.6) with an additional constraint on the average interference:

$$\begin{aligned}
\underset{\{i^{*(t)}\},\{\mathbf{P}^{(t)}\}}{\text{minimize}} \quad & \textstyle\sum_{i=1}^{N} \bar{W}_i \\
\text{subject to} \quad & \textstyle\sum_{i=1}^{N} P_i^{(t)} g_i^{(t)} \leq I_{\text{inst}} \quad , \quad \forall t \geq 1 \\
& I \leq I_{\text{avg}} \\
& \bar{W}_i \leq d_i \\
& P_i^{(t)} \leq P_{\max} \;\; , \;\; \forall i \in \mathcal{N} \;\text{ and }\; \forall t \geq 1, \\
& \textstyle\sum_{i=1}^{N} \mathbb{1}\left(P_i^{(t)}\right) \leq 1 \quad , \quad \forall t \geq 1,
\end{aligned} \tag{3.8}$$

We notice that problems (3.6) and (3.8) are joint power allocation and scheduling problems where the objective function and constraints are expressed in terms of asymptotic time averages and cannot be solved by conventional optimization techniques. The next section proposes low complexity update policies and proves their optimality.

3.3 Proposed Power Allocation and Scheduling Algorithm

We solve problems (3.6) and (3.8) by proposing online joint scheduling and power allocation policies that dynamically update the scheduling and the power allocation. We show that these policies have performances that come arbitrarily close to being optimal. That is, we can achieve a sum of the average delays arbitrarily close to its optimal value depending on some control parameter V.

We first discuss the idea behind our policies. Then we present the proposed policy for each problem, (3.6) and (3.8), separately.

3.3.1 Frame-Based Policy

The idea behind the policies that solve (3.6) and (3.8) is to divide time into frames where frame k consists of a random number T_k time slots and update the power allocation and scheduling at the beginning of each frame. Where each frame begins and ends is specified by idle periods and will be precisely defined later in this section. During frame k, SUs are scheduled according to some priority list $\boldsymbol{\pi}(k)$ and each SU is assigned some power to be used when it is assigned the channel. The priority list and the power functions are fixed during the entire frame k and are found at the beginning of frame k based on the history of the SUs' time-averaged delays and, in the case of (3.8), the PU's suffered interference up to the end of frame $k-1$.

We define $\boldsymbol{\pi}(k) \triangleq [\pi_1(k), \cdots, \pi_N(k)]^T$ where $\pi_j(k)$ is the index of the SU who is given the jth priority during frame k. Given $\boldsymbol{\pi}(k)$, the scheduler becomes a priority scheduler with preemptive-resume priority queuing discipline [56, pp. 205]. The idea of dividing time into frames and assigning fixed priority lists for each frame was also used in [21]. Lemma 1 of [21] proves that restricting the scheduling algorithm to frame-based preemptive-resume priority lists does not result in any loss of optimality.

Frame k consists of $T_k \triangleq |\mathcal{F}(k)|$ consecutive time-slots, where $\mathcal{F}(k)$ is the set containing the indices of the time slots belonging to frame k (see Fig. 3.2). Each frame consists of exactly one *idle period* followed by exactly one *busy period*, both are defined next.

Definition 1. *An idle period is the time interval formed by the consecutive time slots where all SUs have empty buffers. An idle period starts with the time slot t_1 following the completion of transmission of the last packet in the system, and ends with a time slot t_2 when one or more of the SUs' buffer receives one a new packet to be transmitted (see Fig. 3.2). In other words, t_1 satisfies $\sum_{i\in\mathcal{N}} Q_i(t_1) = 0$ and $\sum_{i\in\mathcal{N}} Q_i(t_1-1) \neq 0$, while t_2 satisfies $\sum_{t=t_1}^{t_2-1} \sum_{i\in\mathcal{N}} Q_i(t) = 0$ and $\sum_{i\in\mathcal{N}} Q_i(t_2) \neq 0$.*

Definition 2. *Busy period is the time interval between two consecutive idle periods.*

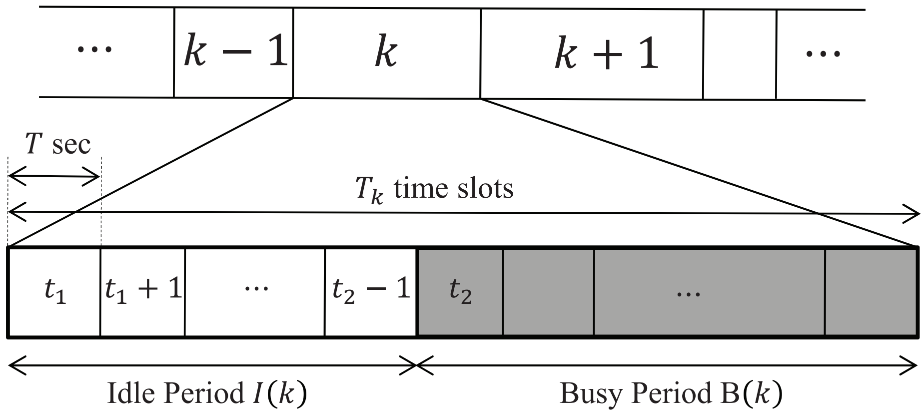

Figure 3.2: Time is divided into frames. Frame k has $T_k \triangleq |\mathcal{F}(k)|$ slots, each is of duration T seconds. Different frames can have different number of time slots.

The duration of the idle period $I(k)$ and busy period $B(k)$ of frame k are random variables, thus $T_k = I(k) + B(k)$ is random as well. Since frames do not overlap, if $t \in \mathcal{F}(k_1)$ then $t \notin \mathcal{F}(k_2)$ as long as $k_1 \neq k_2$. Our goal in this chapter is to choose, at the beginning of each frame k, the best priority list $\boldsymbol{\pi}(k)$ as well as the best power allocation policy for each SU so that the system has an optimal average delay performance satisfying the constraints in (3.6) or (3.8). An equivalent equation for the average delay equation in (3.5) is

$$\bar{W}_i \triangleq \lim_{K\to\infty} \frac{\mathbb{E}\left[\sum_{k=0}^{K}\left(\sum_{j\in\mathcal{A}_i(k)} W_i^{(j)}\right)\right]}{\mathbb{E}\left[\sum_{k=0}^{K}|\mathcal{A}_i(k)|\right]} \tag{3.9}$$

where $\mathcal{A}_i(k) \triangleq \cup_{t\in\mathcal{F}(k)}\mathcal{A}_i^{(t)}$ is the set of all packets that arrive at SU i's buffer during frame k. We note that the long-term average delay $\bar{W}_i$ in (3.9) depends on the chosen priority lists as well as the power allocation policy, in all frames $k \geq 0$.

3.3.2 Satisfying Delay Constraints

In order to guarantee a feasible solution satisfying the delay constraints in problems (3.6) and (3.8), we set up a "virtual queue" associated with each delay constraint

$\bar{W}_i \leq d_i$. The virtual queue for SU i at frame k is given by

$$Y_i(k+1) \triangleq \left(Y_i(k) + \sum_{j \in \mathcal{A}_i(k)} \left(W_i^{(j)} - r_i(k) \right) \right)^+ \tag{3.10}$$

where $r_i(k) \in [0, d_i]$ is an auxiliary random variable, that is to be optimized over and $Y_i(0) \triangleq 0$, $\forall i$. We define $\mathbf{Y}(k) \triangleq [Y_1(k), \cdots, Y_N(k)]^T$. Equation (3.10) is calculated at the end of frame $k-1$ and represents the amount of delay exceeding the delay bound d_i for SU i up to the beginning of frame k. We first give the following definition, then state a lemma that gives a sufficient condition on $Y_i(k)$ for the delay of SU i to satisfy $\bar{W}_i \leq d_i$.

Definition 3. *A random sequence* $\{Y_i(k)\}_{k=0}^{\infty}$ *is mean rate stable if and only if* $\lim_{K\to\infty} \mathbb{E}\left[Y_i(K)\right]/K = 0$ *holds.*

Lemma 3. *If* $\{Y_i(k)\}_{k=0}^{\infty}$ *is mean rate stable, then the time-average delay of SU i satisfies* $\bar{W}_i \leq d_i$.

Proof. Removing the $(\cdot)^+$ sign from equation (3.10) yields

$$Y_i(k+1) \geq Y_i(k) + \sum_{j \in \mathcal{A}_i(k)} \left(W_i^{(j)} - r_i(k) \right). \tag{3.11}$$

Summing inequality (3.11) over $k = 0, \cdots K-1$ and noting that $Y_i(0) = 0$ gives

$$Y_i(K) \geq \sum_{k=0}^{K-1} \left(\sum_{j \in \mathcal{A}_i(k)} W_i^{(j)} \right) - \sum_{k=0}^{K-1} \left(r_i(k) |\mathcal{A}_i(k)| \right). \tag{3.12}$$

Taking the $\mathbb{E}\left[\cdot\right]$ then dividing by $\mathbb{E}\left[\sum_{k=0}^{K-1} |\mathcal{A}_i(k)|\right]$ gives

$$\frac{\mathbb{E}\left[\sum_{k=0}^{K-1}\left(\sum_{j\in\mathcal{A}_i(k)} W_i^{(j)}\right)\right]}{\mathbb{E}\left[\sum_{k=0}^{K-1}|\mathcal{A}_i(k)|\right]} \leq \frac{\mathbb{E}\left[Y_i(K)\right]}{K} \frac{K}{\mathbb{E}\left[\sum_{k=0}^{K-1}|\mathcal{A}_i(k)|\right]} + \frac{\sum_{k=0}^{K-1}\mathbb{E}\left[|\mathcal{A}_i(k)| r_i(k)\right]}{\sum_{k=0}^{K-1}\mathbb{E}\left[|\mathcal{A}_i(k)|\right]}. \tag{3.13}$$

Replacing $r_i(k)$ by its upper bound d_i, taking the limit as $K \to \infty$ then using the mean rate stability definition and equation (3.9) completes the proof. $\square$

Lemma 7 provides a condition on the virtual queue $\{Y_i(k)\}_{k=0}^{\infty}$ so that SU i's average delay constraint $\bar{W}_i \leq d_i$ in (3.6) and (3.8) is satisfied. That is, if the proposed joint power allocation and scheduling policy results in a mean rate stable $\{Y_i(k)\}_{k=0}^{\infty}$, then $\bar{W}_i \leq d_i$. For both problems, the proposed policy depends on the Lyapunov optimization where the goal is to choose the joint scheduling and power allocation policy that minimizes the drift-plus-penalty. In Section 3.3.3 (Section 3.3.4) we will show that if problem (3.6) (problem (3.8)) is feasible, then the proposed policy guarantees mean rate stability for the queues $\{Y_i(k)\}_{k=0}^{\infty}$.

3.3.3 Algorithm for Instantaneous Interference Constraint

We now propose the *Delay Optimal with Instantaneous Interference Constraint* (*DOIC*) policy that solves problem (3.6). This policy is executed at the beginning of each frame k for finding $\mathbf{P}^{(t)}$ as well as the optimum list $\boldsymbol{\pi}(k)$, given some prespecified control parameter V. Given some fixed constant P define the random variable $R_i(P)$ as

$$R_i(P) \triangleq \log\left(1 + \min\left(\frac{I_{\text{inst}}}{g_i^{(t)}}, P\right)\gamma_i^{(t)}\right), \tag{3.14}$$

which is a special case of $R_i^{(t)}$ given in (3.1), and define $\mu_i(P) \triangleq \mathbb{E}\left[R_i(P)\right]/L$ where the expectation is taken over $g_i^{(t)}$ and $\gamma_i^{(t)}$. The *DOIC* policy is as follows.

DOIC Policy (executed at the beginning of frame k):

1. The BS sorts the SUs according to the descending order of $Y_i(k)\mu_i(P_{\max})$. The sorted list is denoted by $\boldsymbol{\pi}(k)$.

2. At the beginning of each slot $t \in \mathcal{F}(k)$ the BS schedules SU i^* that has the highest priority in the list $\boldsymbol{\pi}(k)$ among those having non-empty buffers.

3. SU i^*, in turn, transmits $M_{i^*}^{(t)}$ packets as dictated by equation (3.2) where

$P_i^{(t)} = 0 \; \forall i \neq i^*$ while $P_{i^*}^{(t)}$ is calculated as

$$P_{i^*}^{(t)} = \min\left(\frac{I_{\text{inst}}}{g_{i^*}^{(t)}}, P_{\max}\right), \tag{3.15}$$

4. At the end of frame k, for all $i \in \mathcal{N}$ the BS updates:

   a) $r_i(k) = d_i$ if $V < Y_i(k)\lambda_i$, and $r_i(k) = 0$ otherwise, and then

   b) $Y_i(k+1)$ via equation (3.10).

Before we discuss the optimality of the *DOIC* in Theorem 3, we define the following quantities. Let $a \triangleq 1 - \Pi_{i=1}^{N}(1-\lambda_i)$ denote the probability of receiving a packet from a user or more at a given time slot, while $C_Y \triangleq \sum_{i=1}^{N} C_{Y_i}$ with $C_{Y_i} \triangleq \sqrt{\mathbb{E}[A^4]\,\mathbb{E}[B^4]} + d_i^2\,\mathbb{E}[A^2]$, where $\mathbb{E}[A^2]$ and $\mathbb{E}[A^4]$ are bounds on the second and fourth moments of the total number of arrivals $\sum_i |\mathcal{A}_i(k)|$ during frame k, respectively, while $\mathbb{E}[B^4]$ is a bound on the fourth moment of the busy period $B(k)$. The finiteness of these moments can be shown to hold if the first four moments of the service time are finite. In Appendix E we show that all the service time moments exist given any distribution for $P_i^{(t)}\gamma_i^{(t)}$.

Theorem 3. *If problem* (3.6) *is feasible, then the proposed* DOIC *policy results in a time average of the SUs' delays satisfying the following inequality*

$$\sum_{i=1}^{N} \bar{W}_i \leq \frac{aC_Y}{V} + \sum_{i=1}^{N} \bar{W}_i^* \tag{3.16}$$

where $\bar{W}_i^*$ is the optimum value of the delay when solving problem (3.6)*, while a and C_Y are as given above. Moreover, the virtual queues $\{Y_i(k)\}_{k=0}^{\infty}$ are mean rate stable $\forall i \in \mathcal{N}$.*

Proof. See Appendix D. □

Theorem 3 says that the objective function of problem (3.6) is upper bounded by the optimum value $\sum_i \bar{W}_i^*$ plus some constant gap that vanishes as $V \to \infty$. Having a vanishing gap means that the *DOIC* policy is asymptotically optimal. Moreover, based on the mean rate stability of the queues $\{Y_i(k)\}_{k=0}^{\infty}$, the set of delay constraints of problem (3.8) is satisfied. The drawback of setting V very large is that the time needed for the algorithm to converge increases. This increase is linear in V [58]. That is, if the number of frames required for the quantity $\sum_i Y_i(k)/(Nk)$ to be less than ϵ (for some $\epsilon > 0$) is $O(K_1)$, then increasing V to βV will require $O(\beta K_1)$ frames for it to be less than ϵ, for any $\beta > 1$. We note that the complexity of the *DOIC* policy is $O(N)$ because calculating $\mu_i(P_{\max})$ is of $O(1)$, while the power is closed-form in (3.15). We note that if problem (3.6) is not feasible, then this is because one of two reasons; either one or more of the constraints is stringent, or otherwise because $\sum_{i=1}^N \lambda_i/\mu_i(P_{\max}) \geq 1$. If it is the former, then the *DOIC* policy will result in a point that is as close as possible to the feasible region. On the other hand, if it is the latter, then we could add an admission controller that limits the average number of packets arriving at buffer i to $\lambda_i(1-\delta)/\left(\sum_{i=1}^N \lambda_i/\mu_i(P_{\max})\right)$ for some $\delta > 0$.

3.3.4 Algorithm for Average Interference Constraint

We now propose the *Delay-Optimal-with-Average-Interference-Constraint* (*DOAC*) policy for problem (3.8). We first give the following useful definitions. Since the scheduling scheme in frame k is a priority scheduling scheme with preemptive-resume queuing discipline, then given the priority list $\boldsymbol{\pi}$ we can write the expected waiting time of all SUs in terms of the average residual time [56, pp. 206] defined as $T_{\pi_j}^{\mathrm{R}} \triangleq \sum_{l=1}^{j} \lambda_{\pi_l} \mathbb{E}\left[s_{\pi_l}^2\right]/2$, where the expectation is taken over $P_{\pi_l}^{(t)}\gamma_{\pi_l}^{(t)}$. The waiting time of

SU π_j that is given the jth priority is [56, pp. 206]

$$W_{\pi_j}\left(P,\mu_{\pi_j}(P),\rho_{\pi_j}(P),\bar{\rho}_{\pi_{j-1}},T^{\mathrm{R}}_{\pi_j}\right)=\frac{1}{\left(1-\bar{\rho}_{\pi_{j-1}}\right)}\left[\frac{1}{\mu_{\pi_j}(P)}+\frac{T^{\mathrm{R}}_{\pi_j}}{\left(1-\bar{\rho}_{\pi_{j-1}}-\rho_{\pi_j}(P)\right)}\right] \tag{3.17}$$

$$\leq\frac{1}{\left(1-\bar{\rho}^{\max}_{\pi_{j-1}}\right)}\left[\frac{1}{\mu_{\pi_j}(P)}+\frac{T^{\mathrm{R}}}{\left(1-\bar{\rho}^{\max}_{\pi_{j-1}}-\rho_{\pi_j}(P)\right)}\right]$$

$$\triangleq W^{\mathrm{up}}_{\pi_j}\left(P,\rho_{\pi_j}(P),\bar{\rho}^{\max}_{\pi_{j-1}},T^{\mathrm{R}}\right), \tag{3.18}$$

where we define $\rho_i(P)\triangleq\lambda_i/\mu_i(P)$, $\bar{\rho}_{\pi_{j-1}}\triangleq\sum_{l=1}^{j-1}\rho_{\pi_l}(P_{\pi_l})$, while T^{R} is an upper bound on $T^{\mathrm{R}}_{\pi_j}$ and is given by $T^{\mathrm{R}}\triangleq\sum_{i=1}^{N}\lambda_i\left(L^2+L\left(1-p_i(P_i^{\min})\right)\right)/p_i^2(P_i^{\min})/2$ with $p_i(P)\triangleq 1-\mathbf{Pr}\left[R_i(P)=0\right]$ and $P_i^{\min}$ is the minimum power satisfying $\rho_i(P_i^{\min})+\sum_{j\neq i}\rho_j(P_{\max})<1$ (see Appendix E for the derivation of T^{R}), while $\bar{\rho}_i^{\max}$ is some upper bound on $\bar{\rho}_i$ that will be defined later. We henceforth drop all the arguments of $W^{\mathrm{up}}_{\pi_j}(P,\bar{\rho}^{\max}_{\pi_{j-1}})$ except P and $\bar{\rho}^{\max}_{\pi_{j-1}}$ and all those of $W_{\pi_j}(P)$ except P.

To track the average interference at the PU up to the end of frame k we set up the following virtual queue that is associated with the average interference constraint in problem (3.8) and is calculated at the BS at the end of frame k.

$$X(k+1)\triangleq\left(X(k)+\sum_{i=1}^{N}\sum_{t\in\mathcal{F}(k)}P_i^{(t)}g_i^{(t)}-I_{\mathrm{avg}}T_k\right)^{+}, \tag{3.19}$$

where the term $\sum_{i=1}^{N}\sum_{t\in\mathcal{F}(k)}P_i^{(t)}g_i^{(t)}$ represents the aggregate amount of interference energy received by the PU due to the transmission of the SUs during frame k. Hence, this virtual queue is a measure of how much the SUs have exceeded the interference constraint above the level I_{avg} that the PU can tolerate. Lemma 4 provides a sufficient condition for the interference constraint of problem (3.8) to be satisfied.

Lemma 4. *If $\{X(k)\}_{k=0}^{\infty}$ is mean rate stable, then the time-average interference received by the PU satisfies $I\leq I_{\mathrm{avg}}$.*

Proof. The proof is similar to that of Lemma 7 and is omitted for brevity. □

Lemma 4 says that if the power allocation and scheduling algorithm results in mean rate stable $\{X(k)\}_{k=0}^{\infty}$, then the interference constraint of problem (3.8) is satisfied.

Before presenting the *DOAC* policy, we first discuss the idea behind it. Theorem 4 will show that the optimum power allocation for SU i is

$$P_i^{(t)} = \min\left(\frac{I_{\text{inst}}}{g_i^{(t)}}, P_i(k)\right), \tag{3.20}$$

where $P_i(k) \in [P_i^{\min}, P_{\max}]$ is a power parameter that is fixed within frame k (i.e. $\forall t \in \mathcal{F}(k)$). Intuitively, a policy that solves problem (3.8) should allocate SU i's power and assign its priority such that SU i's expected delay and the expected interference to the PU is minimized. The *DOAC* policy is defined as the policy that selects the power parameter vector $\mathbf{P}(k) \triangleq [P_1(k), \cdots, P_N(k)]^T$ jointly with the priority list $\boldsymbol{\pi}(k)$ that minimizes $\Psi \triangleq \sum_{j=1}^{N} \psi_{\pi_j}(P_{\pi_j}(k), \bar{\rho}_{\pi_{j-1}}^{\max})$ where

$$\psi_{\pi_j}(P, \bar{\rho}_{\pi_{j-1}}^{\max}) \triangleq \psi_{\pi_j}^{\text{D}}(P, \bar{\rho}_{\pi_{j-1}}^{\max}) + \psi_{\pi_j}^{\text{I}}(P), \quad with \tag{3.21}$$

$$\psi_{\pi_j}^{\text{D}}(P, \bar{\rho}_{\pi_{j-1}}^{\max}) \triangleq Y_{\pi_j}(k)\lambda_{\pi_j} W_{\pi_j}^{\text{up}}(P, \bar{\rho}_{\pi_{j-1}}^{\max}), \quad while \tag{3.22}$$

$$\psi_{\pi_j}^{\text{I}}(P) \triangleq X(k)\rho_{\pi_j}(P)P\bar{g}_{\pi_j}. \tag{3.23}$$

The function $\psi_{\pi_j}^{\text{D}}(P, \bar{\rho}_{\pi_{j-1}}^{\max})$ (and $\psi_{\pi_j}^{\text{I}}(P)$) represents the amount of delay (interference) that SU π_j is expected to experience (to cause to the PU) during frame k.

The brute search of $\mathbf{P}(k)$ and $\boldsymbol{\pi}(k)$ that minimizes Ψ is exponentially high. To minimize Ψ in a computationally efficient way, we need the functions $\psi_{\pi_j}(P_{\pi_j}(k), \bar{\rho}_{\pi_{j-1}}^{\max})$ to become decoupled for all $j \in \mathcal{N}$. That is, we want $\psi_{\pi_j}(P_{\pi_j}(k), \bar{\rho}_{\pi_{j-1}}^{\max})$ not to depend on $P_{\pi_l}(k)$ as long as $l \neq j$. Hence, we set the function $\bar{\rho}_{\pi_{j-1}}^{\max}$ to some function that does not depend on the optimization power variables $P_{\pi_l}(k)$ for all $l \leq j-1$ but otherwise on some other fixed parameters. We need to choose these parameters such that the bound

$$\bar{\rho}_{\pi_{j-1}}^{\max} \geq \bar{\rho}_{\pi_{j-1}} \triangleq \sum_{l=1}^{j-1} \rho_{\pi_l}(P_{\pi_l}) \tag{3.24}$$

is satisfied. Thus, these functions, are given by

$$\bar{\rho}_{\pi_{j-1}}^{\max} \triangleq \sum_{l=1}^{j-1} \rho_{\pi_l}\left(P_{\pi_l}^{\bar{\rho}^{\max}}\right), \tag{3.25}$$

where

$$P_{\pi_l}^{\bar{\rho}^{\max}} \triangleq \arg\min_{P} \psi_{\pi_l}\left(P, \bar{\rho}_{\pi_{l-1}}^{\max}\right). \tag{3.26}$$

With $\bar{\rho}_{\pi_{j-1}}^{\max}$ given by (3.25), $\psi_{\pi_j}(P_{\pi_j}(k), \bar{\rho}_{\pi_{j-1}}^{\max})$ is a function in $P_{\pi_j}(k)$ only. Before we show that the choice of (3.25) and (3.26) guarantees that (3.24) is satisfied, we note that (3.25) dictates that in order to find $\bar{\rho}_{\pi_{j-1}}^{\max}$ we need to find $P_{\pi_l}^{\bar{\rho}^{\max}}$ for all $l < j-1$. Hence, we find $P_{\pi_j}^{\bar{\rho}^{\max}}$ recursively starting from $j = 1$ at which $\bar{\rho}_{\pi_0}^{\max} = 0$ by definition. We will show that $\bar{\rho}_{\pi_j}^{\max}$ is an upper bound on $\bar{\rho}_{\pi_j}$ in the following lemma.

Lemma 5. *Given some priority list $\boldsymbol{\pi}(k)$, for any user $\pi_j \in \mathcal{N}$ the function $\bar{\rho}_{\pi_j}$ evaluated at the power vector $\mathbf{P}^{\bar{\rho}}$ which is the power vector that minimizes $\sum_{j=1}^{N} \psi_{\pi_j}(P_{\pi_j}, \bar{\rho}_{\pi_{j-1}})$, is upper bounded by $\bar{\rho}_{\pi_j}^{\max}$. Namely,*

$$\bar{\rho}_{\pi_j}\Big|_{\mathbf{P}^{\bar{\rho}}} \leq \bar{\rho}_{\pi_j}^{\max}\Big|_{\mathbf{P}^{\bar{\rho}^{\max}}} \tag{3.27}$$

where

$$\mathbf{P}^{\bar{\rho}} \triangleq \arg\min_{\mathbf{P}} \sum_{j=1}^{N} \psi_{\pi_j}(P_{\pi_j}, \bar{\rho}_{\pi_{j-1}}) \tag{3.28}$$

while

$$\mathbf{P}^{\bar{\rho}^{\max}} \triangleq \arg\min_{\mathbf{P}} \sum_{j=1}^{N} \psi_{\pi_j}(P_{\pi_j}, \bar{\rho}_{\pi_{j-1}}^{\max}) \tag{3.29}$$

Proof. We first argue that $P_{\pi_j}^{\bar{\rho}} \geq P_{\pi_j}^{\bar{\rho}^{\max}}$ for any $j \in \mathcal{N}$. Then we show that $\bar{\rho}_{\pi_j}$ is decreasing in P_{π_l} for all $l \leq j$ which completes the proof.

From (3.29), we have

$$P_{\pi_j}^{\bar{\rho}^{\max}} = \underset{P_{\pi_j} \leq P_{\max}}{\arg\min} \sum_{l=1}^{N} \psi_{\pi_l} \left(P_{\pi_l}, \bar{\rho}_{\pi_{l-1}}^{\max} \right). \tag{3.30}$$

But since $\bar{\rho}_{\pi_j}^{\max}$ is not a function in P_{π_l} except if $l = j$, then we can remove the summation in (3.30) and set $l = j$ yielding

$$P_{\pi_j}^{\bar{\rho}^{\max}} = \underset{P_{\pi_j} \leq P_{\max}}{\arg\min} \left[\psi_{\pi_j}^{\mathrm{I}} \left(P_{\pi_j} \right) + \psi_{\pi_j}^{\mathrm{D}} \left(P_{\pi_j}, \bar{\rho}_{\pi_{j-1}}^{\max} \right) \right] \tag{3.31}$$

Moreover, from (3.28), we have

$$P_{\pi_j}^{\bar{\rho}} = \underset{P_{\pi_j} \leq P_{\max}}{\arg\min} \sum_{l=1}^{N} \psi_{\pi_l} \left(P_{\pi_l}, \bar{\rho}_{\pi_{l-1}} \right) \tag{3.32}$$

$$= \underset{P_{\pi_j} \leq P_{\max}}{\arg\min} \left[\psi_{\pi_j}^{\mathrm{I}} \left(P_{\pi_j} \right) + \sum_{l=j}^{N} \psi_{\pi_l}^{\mathrm{D}} \left(P_{\pi_l}, \bar{\rho}_{\pi_{l-1}} \right) \right] \tag{3.33}$$

If $\psi_{\pi_j}^{\mathrm{I}} \left(P_{\pi_j} \right)$ is non increasing in P_{π_j} over its entire domain, then the optimum solution for (3.33) is $P_{\pi_j}^{\bar{\rho}} = P_{\max}$, which is the same as the optimum solution of $P_{\pi_j}^{\bar{\rho}^{\max}}$ from (3.31). Hence, we continue the proof assuming that there exists a region in the domain of P_{π_j} where $\psi_{\pi_j}^{\mathrm{I}} \left(P_{\pi_j} \right)$ is increasing.

Since $W_{\pi_j}^{\mathrm{up}}(P_{\pi_j}, \bar{\rho}_{\pi_{j-1}})$ is decreasing in P_{π_j}, $\psi_{\pi_j}^{\mathrm{D}} \left(P_{\pi_j}, \bar{\rho}_{\pi_{j-1}} \right)$ is also decreasing in P_{π_j}. Hence, the summation in (3.33) is decreasing in P_{π_j}. At the same time, $\psi_{\pi_j}^{\mathrm{I}} \left(P_{\pi_j} \right)$ is increasing in P_{π_j}. Thus, there are two forces in the objective of (3.33) that determine the optimum value of $P_{\pi_j}^{\bar{\rho}}$; the first one is represented by $\psi_{\pi_j}^{\mathrm{I}} \left(P_{\pi_j} \right)$ and is in favor of decreasing it, while the second is the summation that is in favor of increasing it. We continue the proof by induction on j. Setting $j = 1$ in (3.33), we can easily see that if we neglect all the terms in the summation except the term when $l = 1$, the value of $P_{\pi_1}^{\bar{\rho}}$ decreases. Namely,

$$P_{\pi_1}^{\bar{\rho}} \geq \underset{P_{\pi_1} \leq P_{\max}}{\arg\min} \left[\psi_{\pi_1}^{\mathrm{I}} \left(P_{\pi_1} \right) + \psi_{\pi_1}^{\mathrm{D}} \left(P_{\pi_1}, \bar{\rho}_{\pi_0} \right) \right] \tag{3.34}$$

$$= P_{\pi_1}^{\bar{\rho}^{\max}}, \tag{3.35}$$

where the last equation follows after setting $j = 1$ in (3.31). From (3.14) we can see that $\bar{\rho}_{\pi_1}$ is decreasing in P_{π_1}. With this in mind, and after using the inequality in (3.35) we get $\bar{\rho}_{\pi_1} \leq \bar{\rho}^{\max}_{\pi_1}$. Setting $j = 2$ in (3.33) and neglecting all terms in the summation except at $j = 2$ yields

$$P^{\bar{\rho}}_{\pi_2} \geq \arg\min_{P_{\pi_2} \leq P_{\max}} \left[\psi^{\mathrm{I}}_{\pi_2}\left(P_{\pi_2}\right) + \psi^{\mathrm{D}}_{\pi_2}\left(P_{\pi_2}, \bar{\rho}_{\pi_1}\right)\right] \tag{3.36}$$

$$\geq \arg\min_{P_{\pi_2} \leq P_{\max}} \left[\psi^{\mathrm{I}}_{\pi_2}\left(P_{\pi_2}\right) + \psi^{\mathrm{D}}_{\pi_2}\left(P_{\pi_2}, \bar{\rho}^{\max}_{\pi_1}\right)\right] \tag{3.37}$$

$$= P^{\bar{\rho}^{\max}}_{\pi_2}. \tag{3.38}$$

Thus we get $\bar{\rho}_{\pi_2} \leq \bar{\rho}^{\max}_{\pi_2}$. Repeating for a general $j \leq N$ and assuming that $\bar{\rho}_{\pi_{j-1}} \leq \bar{\rho}^{\max}_{\pi_{j-1}}$, we get $P^{\bar{\rho}}_{\pi_2} \geq P^{\bar{\rho}^{\max}}_{\pi_2}$ yielding $\bar{\rho}_{\pi_j} \leq \bar{\rho}^{\max}_{\pi_j}$ which completes the proof. $\square$

Lemma 5 states that we can replace $\bar{\rho}_{\pi_j}$ by $\bar{\rho}^{\max}_{\pi_j}$ to upper bound (3.17) with (3.18). $\bar{\rho}^{\max}_{\pi_j}$ has an advantage over $\bar{\rho}_{\pi_j}$ (and hence $\psi_{\pi_j}\left(P_{\pi_j}, \bar{\rho}^{\max}_{\pi_{j-1}}\right)$ over $\psi_{\pi_j}\left(P_{\pi_j}, \bar{\rho}_{\pi_{j-1}}\right)$) which is that it is not a function in P_{π_l} for $l \neq j$. This decouples the power search optimization problem to N one-dimensional searches.

After reducing the search complexity of the power vector, we reduce the search complexity of the priority list from $N!$ to 2^N. To do this, we use the dynamic programming illustrated in Algorithm 2 that solves $\min_{\boldsymbol{\pi}(k),\mathbf{P}(k)} \Psi$. Its search complexity is of $O(MN2^N)$ where M is the number of iterations in a one-dimensional search, while $O(1)$ is the complexity of calculating Ψ for a given priority list $\boldsymbol{\pi}(k)$ and a given power vector $\mathbf{P}(k)$. Compared to the complexity of $O(M^N \cdot N!)$ which is that of the N-dimensional power search along with the brute-force of all $N!$ permutations of priority list $\boldsymbol{\pi}(k)$, this is a large complexity reduction. However, the $O(MN2^N)$ is still high if N was large. Finding an optimal algorithm with a lower complexity is extremely difficult since the scheduling and power control problem are coupled. In other words, in order to find the optimum scheduler we need to know the optimum

Algorithm 2 *DOAC-Pow-Alloc*: Optimization-problem-solution algorithm called by the *DOAC* policy at the beginning of frame k to solve for $\mathbf{P}^*(k)$ as well as $\boldsymbol{\pi}^*(k)$.

1: Define $\mathcal{S}$ as the set of all sets formed of all subsets of $\mathcal{N}$ and define the auxiliary functions

$$\begin{aligned}
&\tilde{\Psi}(\cdot,\cdot) : \mathcal{N} \times \mathcal{S} \to \mathbb{R}^+ \\
&\tilde{\rho}(\cdot) : \mathcal{S} \to [0,1], \\
&\tilde{\mathbf{S}}(\mathcal{X}) : \mathcal{S} \to \mathcal{N}^{|\mathcal{X}|}, \\
&\tilde{\mathbf{P}}(\mathcal{X}) : \mathcal{S} \to [0, P_{\max}]^{|\mathcal{X}|}, \\
&\bar{P}(\cdot,\cdot) : \mathcal{S} \times \mathcal{N} \to [0, P_{\max}].
\end{aligned}$$

2: Initialize $\tilde{\Psi}(0,\cdot) = 0$, $\tilde{\rho}(\phi) = 0$, $\tilde{\mathbf{S}}(\phi) = [\,]$ and $\tilde{\mathbf{P}}(\phi) = [\,]$, where ϕ is the empty set.

3: **for** $i = 1, \cdots, N$ **do**

4: In stage i, the first i priorities have been assigned to i users. The corresponding priority list is denoted $[\pi_1, \cdots, \pi_i]$. In stage i we have $\binom{N}{i}$ states each corresponds to a set j formed from all possible combinations of i elements chosen from the set $\mathcal{N}$. We calculate $\tilde{\Psi}(i,j)$ associated with each state j in terms of $\tilde{\Psi}(i-1,\cdot)$ obtained in stage $i-1$ as follows.

5: **for** $j \in$ all possible i-element sets **do**

6: At state $j \triangleq \{\pi_1, \cdots, \pi_i\}$, we have i transitions, each connects it to state j' in stage $i-1$, where $j' \triangleq j \backslash l$ with $l \in j$. Find the power associated with each transition $l \in j$ denoted

$$\bar{P}(j,l) \triangleq \arg\min_{P} \psi_l(P, \tilde{\rho}(j \backslash l)) \tag{3.39}$$

7: Set

$$\begin{aligned}
l^* &= \arg\min_{l \in j} \tilde{\Psi}\left(i-1, j \backslash l\right) + \psi_l\left(\bar{P}(j,l), \tilde{\rho}(j \backslash l)\right), \\
\tilde{\Psi}(i,j) &= \tilde{\Psi}(i-1, j \backslash l^*) + \psi_{l^*}\left(\bar{P}(j,l^*), \tilde{\rho}(j \backslash l^*)\right), \\
\tilde{\rho}(j) &= \tilde{\rho}\left(j \backslash l^*\right) + \rho\left(\bar{P}(j,l^*)\right), \\
\tilde{\mathbf{S}}(j) &= \left[\tilde{\mathbf{S}}\left(j \backslash l^*\right), l^*\right]^T, \\
\tilde{\mathbf{P}}(j) &= \left[\tilde{\mathbf{P}}\left(j \backslash l^*\right), \bar{P}(j,l^*)\right]^T.
\end{aligned}$$

8: **end for**

9: **end for**

10: Set $\boldsymbol{\pi}^*(k) = \tilde{\mathbf{S}}\left(\mathcal{N}\right)$ and $\mathbf{P}^*(k) = \tilde{\mathbf{P}}\left(\mathcal{N}\right)$.

power vector and vice versa. In Section 3.3.5 we propose a sub-optimal policy with a very low complexity and little degradation in the delay performance. We now present the *DOAC* policy that the BS executes at the beginning of frame k.

***DOAC* Policy** (executed at the beginning of frame k):

1. The BS executes *DOAC-Pow-Alloc* in Algorithm 2 to find the optimum power parameter vector $\mathbf{P}^*(k) \triangleq [P_1^*(k), \cdots, P_N^*(k)]^T$ as well as the optimum priority list $\boldsymbol{\pi}^*(k) \triangleq [\pi_1^*(k), \cdots, \pi_N^*(k)]^T$ that will be used during frame k.

2. The BS broadcasts the vector $\mathbf{P}^*(k)$ to the SUs.

3. At the beginning of each slot $t \in \mathcal{F}(k)$, the BS schedules SU i^* that has the highest priority in the list $\boldsymbol{\pi}^*(k)$ among those having non-empty buffers.

4. SU $i^{*(t)}$, in turn, transmits $M_{i^{*(t)}}^{(t)}$ bits as dictated by equation (3.2) where $P_i^{(t)} = 0$ for all $i \neq i^{*(t)}$ while $P_{i^{*(t)}}^{(t)}$ is given by equation (3.20).

5. At the end of frame k, for all $i \in \mathcal{N}$ the BS updates:

    a) $r_i(k) = d_i$ if $V < Y_i(k)\lambda_i$, and $r_i(k) = 0$ otherwise,

    b) $X(k+1)$ via equation (3.19),

    c) $Y_i(k+1)$ via equation (3.10), $\forall i \in \mathcal{N}$.

Define $C_X \triangleq \left(P_{\max}^2 g_{\max}^2 + I_{\text{avg}}^2\right)\left((1-a)(2+a) + \mathbb{E}\left[B^2\right] + 2\,\mathbb{E}\left[B\right](a-a^2)\right)/a^2$ and $C \triangleq C_Y + C_X$ where $\mathbb{E}\left[B\right]$ is a bound on the mean of $B(k)$. It can be shown that $\mathbb{E}\left[B\right]$ and $\mathbb{E}\left[B^2\right]$ are finite if the first two moments of the service time are finite (see Appendix E for a proof of the finiteness of the service time moments). Thus, C_X is finite. Next, we state Theorem 4 that discusses the optimality of the *DOAC* policy.

Theorem 4. *When the BS executes the* DOAC *policy, the time average of the SUs' delays satisfy the following inequality*

$$\sum_{i=1}^{N} \bar{W}_i \leq \frac{aC}{V} + \sum_{i=1}^{N} \bar{W}_i^* \tag{3.40}$$

where $\bar{W}_i^*$ *is the optimum value of the delay when solving problem* (3.8)*. Moreover, the virtual queues* $\{X(k)\}_{k=0}^{\infty}$ *and* $\{Y_i(k)\}_{k=0}^{\infty}$ *are mean rate stable* $\forall i \in \mathcal{N}$*.*

Proof. See Appendix F. □

Theorem 4 says that the objective function of problem (3.8) is upper bounded by the optimum value $\sum_i \bar{W}_i^*$ plus some constant gap that vanishes as $V \to \infty$. Having a vanishing gap means that the *DOAC* policy is asymptotically optimal. Moreover, based on the mean rate stability of $\{X(k)\}_{k=0}^{\infty}$ and $\{Y_i(k)\}_{k=0}^{\infty}$, the interference and delay constraints of problem (3.8) are satisfied.

3.3.5 Near-Optimal Low Complexity Algorithm for Average Interference Problem

As seen in the *DOAC* policy, the complexity of finding the optimal power vector and priority list can be high when the number of SUs N is large. This is mainly due to the large complexity of Algorithm 2. In this subsection we propose a suboptimal solution with an extreme reduction in complexity and with little degradation in the performance. This solution solves for the power allocation and scheduling algorithm, thus it replaces the Algorithm 2.

The challenges in Algorithm 2 are three-fold. First finding the priority list (scheduling problem) requires the search over $N!$ possibilities. Second, even with a genie-aided knowledge of the optimum list, we still have to carry-out N one-dimensional searches to find $\mathbf{P}^*(k)$ (power control problem). Third, the scheduling and power control problems are coupled. We tackle the latter two challenges first, by finding a low-complexity power allocation policy that is independent of the scheduling algorithm. Then we use the $c\mu$ rule [59] to find the priority list. The $c\mu$ rule is a policy that gives the priority list that minimizes the quantity $\sum_{i=1}^{N} Y_i(k)\lambda_i W_i(P_i(k))$, given some power allocation vector $\mathbf{P}(k)$.

Define $P_{\min}$ to be the minimum power that satisfies $\sum_{j=1}^{N} \rho_{\pi_j}(P_{\min}) < 1$. Intuitively, if, for some $\pi_j \in \mathcal{N}$, $X(k) \gg Y_{\pi_j}(k)$ then $P^*_{\pi_j}(k)$ is expected to be close to $P_{\min}$ since the interference term $\psi^{\mathrm{I}}_{\pi_j}(P)$ dominates over $\psi^{\mathrm{D}}_{\pi_j}(P)$ in the π_jth term of the summation in equation (3.21). On the other hand, if $X(k) \ll Y_{\pi_j}(k)$ then $P^*_{\pi_j}(k) \approx P_{\max}$. We propose the following power allocation policy for SU $\pi_j \in \mathcal{N}$

$$\hat{P}_{\pi_j}(k) = \begin{cases} P_{\min} \text{ if } X(k) > Y_{\pi_j}(k) \\ P_{\max} \text{ otherwise.} \end{cases} \tag{3.41}$$

We can see that the power allocation policy in (3.41) does not depend on the position of SU i in the priority list as opposed to Algorithm 2 which requires the knowledge of SU π_j's priority position. In other words, $\hat{P}_{\pi_j}(k)$ is a function of π_j but it is not a function of j. Before proposing the scheduling policy, we note the following two properties. First, with a genie-aided knowledge of the power $\mathbf{P}^*(k)$, and when $X(k) = 0$, the solution to the minimization problem $\min_{\boldsymbol{\pi}} \Psi$ is given by the $c\mu$ rule [59] that sorts the SUs according to the descending order of $Y_{\pi_j}(k)\mu_{\pi_j}(\hat{P}_{\pi_j}(k))$. Second, with a genie-aided knowledge of the power $\mathbf{P}^*(k)$, and when $Y_{\pi_j}(k) = 0\ \forall \pi_j \in \mathcal{N}$, any sorting order would not affect the objective function Ψ.

The two-step scheduling and power allocation algorithm that we propose is 1) allocate the power vector $\mathbf{P}(k)$ according to (3.41), then 2) assign priorities to the SUs in a descending order of $Y_{\pi_j}(k)\mu_{\pi_j}(\hat{P}_{\pi_j}(k))$ (the $c\mu$ rule). The complexity of this algorithm is that of sorting N numbers, namely $O(N\log(N))$. This is a very low complexity if compared to that of the $DOAC$ policy of $O(MN \cdot N!)$. In Section 4.6

we will demonstrate that this huge reduction of complexity causes little degradation to the delay performance.

3.4 Achievable Rate Region of the *DOAC*

We have shown that the *DOAC* policy is delay optimal. In this section we show how much of the capacity region this policy achieves. We also present different scenarios where the *DOAC* policy achieves the whole capacity region, hence becoming both throughput optimal and delay optimal at the same time.

Theorem 1 in [58, pp. 52] explicitly states the capacity region in the case of an instantaneous power constraint. In general this capacity region is strictly convex. A simple example of this capacity region is shown in Fig. 4.5 for a 2-user case with channel gains $\gamma_i^{(t)} \in \{0,1\}$ while $g_i^{(t)} = 0$, for all $i = 1, 2$. The next lemma presents the rate region that the *DOAC* achieves.

Lemma 6. *Under the* DOAC *policy, the queues of all users will be stable if and only if the arrival rate vector satisfies* $\sum_{i \in \mathcal{N}} \rho_i(P_{\max}) < 1$ *with strict inequality.*

Proof. If $\sum_{i \in \mathcal{N}} \rho_i(P_{\max}) \geq 1$, then for any $\pi \in \mathcal{P}$ we will have $\bar{W}_{\pi_N} = \infty$ from (3.5). Thus, the queue of at least one of the users will build up. Moreover, if the inequality $\sum_{i \in \mathcal{N}} \rho_i(P_{\max}) < 1$ holds, we will have $\bar{W}_i < \infty$ for all $i \in \mathcal{N}$. Little's law completes the proof. □

The achievable region provided in Lemma 8 is shown in Fig. 4.5 for the 2-user case. This is a straight line intersecting the two axes at $(\mu_1(P_{\max}), 0)$ and $(0, \mu_2(P_{\max}))$, respectively. Although, in general, this rate region lies strictly inside the capacity region, there are cases where the two regions coincide. Before presenting two of these examples, we note that in these cases the *DOAC* is delay optimal and throughput optimal at the same time.

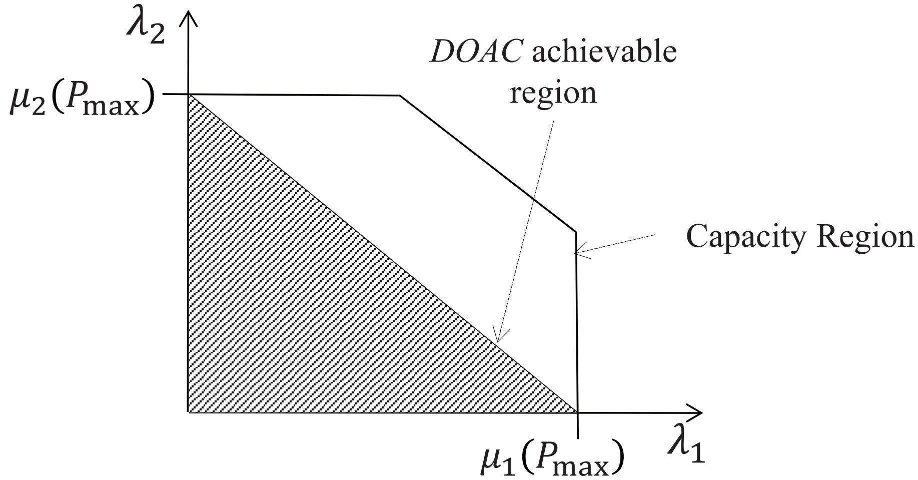

Figure 3.3: Capacity region for a 2-user case.

Example 1 (Unknown channel gain): If all SUs are not able to estimate the gain of their direct channel to their BS, then each SU would be transmitting with a fixed rate that corresponds to the minimum non-zero channel gain $\gamma_i^{\min} \triangleq \min_{\gamma_i^{(t)} \neq 0} \gamma_i^{(t)}$. Hence the capacity region shrinks [60, pp. 115] to be the region bounded by the hyper plane intersecting the ith axis at the point $[0, \cdots, \mu_i^{\min}, 0, \cdots]^T$ where

$$\mu_i^{min} \triangleq \frac{\log\left(1 + P_{\max}\gamma_i^{\min}\right)\left(1 - \mathbf{Pr}\left[\gamma_i^{(t)} = 0\right]\right)}{L}, \tag{3.42}$$

thus coinciding with the *DOAC* achievable rate region.

Example 2 (Non-fading channel): When we have a non-fading channel, each SU transmits with a fixed rate equals $\log(1 + P_{\max})$ bits per slot. Hence the capacity region becomes the region in the first quadrant that is bounded by the hyper plane intersecting the each axis at $\log(1 + P_{\max})/L$, thus coinciding with the *DOAC* achievable rate region.

3.5 Performance Under Channel Estimation Errors

In this section, we present the solution of the system under channel estimation errors. We assume that SU i estimates $\gamma_i^{(t)}$ and $g_i^{(t)}$ with $\alpha\%$ error relative to its actual value, where $\alpha > 0$ represents the percentage of maximum deviation from the true value. This is a good model when the source of channel errors is mainly due to quantization. The observed values satisfy

$$\left(1-\frac{\alpha}{2}\right)\gamma_i^{(t)} \leq \gamma_i^{\text{err}}(t) \leq \left(1+\frac{\alpha}{2}\right)\gamma_i^{(t)}, \tag{3.43}$$

$$\left(1-\frac{\alpha}{2}\right)g_i^{(t)} \leq g_i^{\text{err}}(t) \leq \left(1+\frac{\alpha}{2}\right)g_i^{(t)}, \tag{3.44}$$

From equation (3.1), in order to prevent outage, we need to consider the worst case scenario for $\gamma_i^{(t)}$. Therefore, we estimate $\gamma_i^{(t)}$ to be

$$\gamma_i^{(t)} = \frac{\gamma_i^{\text{err}}(t)}{1+\frac{\alpha}{2}}. \tag{3.45}$$

Although this is a worst case estimation of $\gamma_i^{(t)}$, we will show through simulations that the reduction in performance is not high even with a relatively high value of α. Similarly, instantaneous interference constraint in equation (3.6) is satisfied using a worst-case estimate of $g_i^{(t)}$ as

$$g_i^{(t)} = \frac{g_i^{\text{err}}(t)}{1-\frac{\alpha}{2}}. \tag{3.46}$$

With the estimated CSI values given by equations (3.45) and (3.46), the two problems of instantaneous and average interference constraint, namely problems (3.6) and (3.8), become functions of the observed CSI values as well as the parameter α. Hence, the two policies *DOIC* and *DOAC* can be used to to solve problems (3.6) and (3.8), respectively, under estimation errors. Section 4.6 simulates this system and shows the performance under this error model.

3.6 Simulation Results

We simulated a system of $N = 2$ SUs. Table 4.1 lists all parameter values for both scenarios; the instantaneous as well as the average interference constraint. We expect SU 1 to have higher average delay in both scenarios. This is because it has a lower

Table 3.1: Simulation Parameter Values

Parameter	Value	Parameter	Value
L	1000 bits per packet	I_{inst}	50
$R_{\max}$	82 bits per slot	$P_{\max}$	100
$\lambda_1 = \lambda_2 = \lambda$	$\lambda \in \{1, \cdots 10\} \times 10^{-3}$ packets/slot	N	2 SUs
$f_{\gamma_i}(\gamma)$	$\exp\left(-\gamma/\bar{\gamma}_i\right)/\bar{\gamma}_i$	α	0.1
$f_{g_i}(g)$	$\exp\left(-g/\bar{g}_i\right)/\bar{g}_i$	ϵ	0.1
$(\bar{\gamma}_1, \bar{\gamma}_2)$	$(2, 4)$	V	10
$(\bar{g}_1, \bar{g}_2)$	$(0.4, 0.2)$	d_2	$40T$

average channel gain and higher interference channel gain compared to those of SU 2. However, the *DOIC* policy can guarantee a bound on this delay using the constraint $\bar{W}_1 \leq d_1$, so that the QoS requirement of SU 1 is satisfied. In our simulations we set $d_1 = 30T$ unless otherwise specified.

3.6.1 Instantaneous Interference

In Figures 3.4 and 3.5 we consider problem (3.6) and assumed perfect knowledge of the direct and interference channel state information (CSI), namely $g_i^{(t)}$ and $\gamma_i^{(t)}$. Fig. 3.4 plots the average per-SU delay $\bar{W}_i$, from equation (3.5), for two cases; the first being the constrained optimization problem where $d_1 = 30T$ while setting d_2 to any arbitrarily high value (we set $d_2 = 40T$), while the second is the unconstrained optimization problem where both d_1 and d_2 are set arbitrarily high (we set $d_1 = d_2 = 40T$). We call it the unconstrained problem because the average delay of both SUs is strictly below $40T$, thus both delay constraints are inactive. The X-axis is the probability of a packet arrival per time slot λ, where $\lambda \triangleq \lambda_1 = \lambda_2$. From Fig. 3.4 we can see a gap, in the unconstrained problem, between the average delay of SU 1 and that of SU 2. Hence, SU 1 suffers from high delay. While for the constrained problem, the *DOIC* policy has forced $\bar{W}_1$ to be smaller than $30T$ for all λ values. This comes at the cost of SU 2's delay. We conclude that the delay constraints in problem (3.6) can force the delay vector of the SUs to take any value as long as it is feasible.

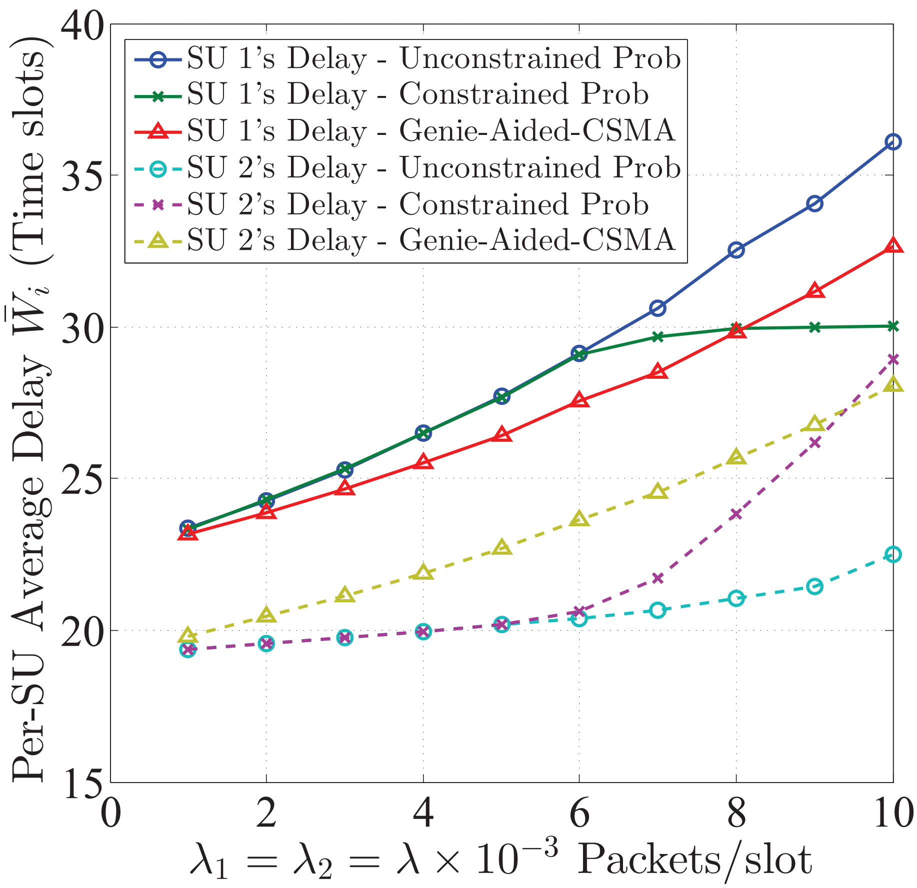

Figure 3.4: Average per-user delay for both the constrained and unconstrained optimization problems

3.6.2 Average Interference

Problem (3.8) differs than problem (3.6) by an additional average interference constraint. This comes at the cost of the sum of average delays of SUs. We simulated the system with $d_1 = 40T$ and compared it to the performance of the *DOIC* policy with $d_1 = 40T$ as well. The sum of average delays of the two SUs is plotted in Fig. 3.6 for both algorithms. The increase in the average delay for the *DOAC* policy is due to adding an additional average interference constraint. However, when comparing the *DOAC* policy to a Carrier-Sense-Multiple-Access (CSMA) scheduling policy we find

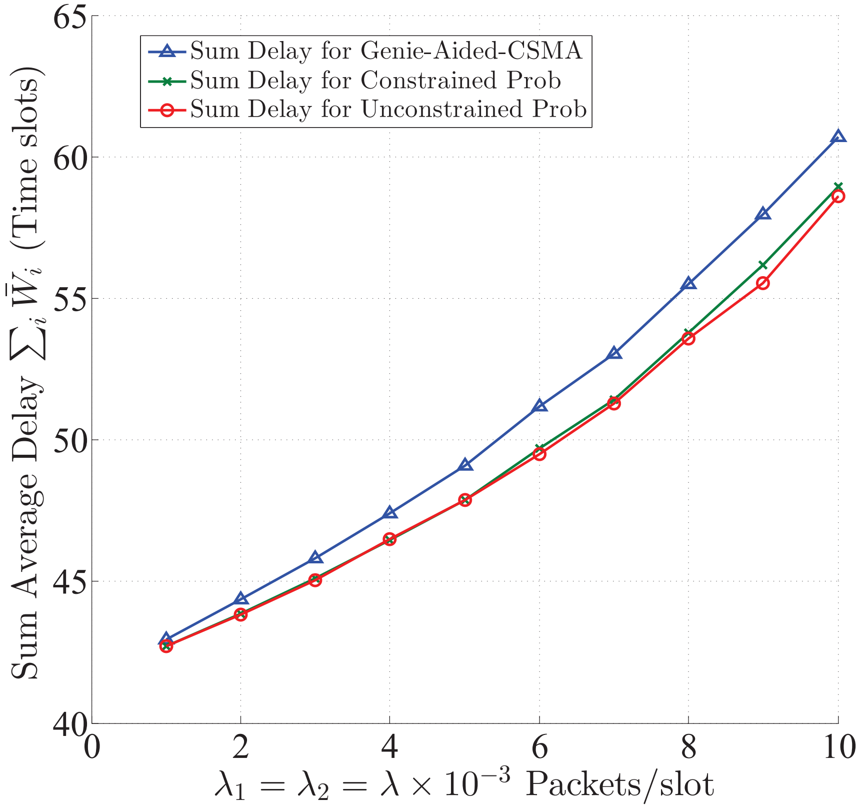

Figure 3.5: Sum of cost functions for the perfect CSI estimates for the *DOIC* policy to solve problem (3.6).

it to have a lower average delay performance. This is because the CSMA allocates the channels randomly uniformly among users and does not prioritize the users based on their delay requirement d_i. On the other hand, the *DOAC* allocates the channels based on the objective of minimizing the sum of average delays. We note that the CSMA policy plotted in Fig. 3.6 uses a "genie-aided" power allocation policy obtained from Algorithm 2. Thus, even when the two algorithms, the CSMA policy and the *DOAC* policy, have the same power allocation policy, the *DOAC* scheduling policy has an improved delay performance over the CSMA policy.

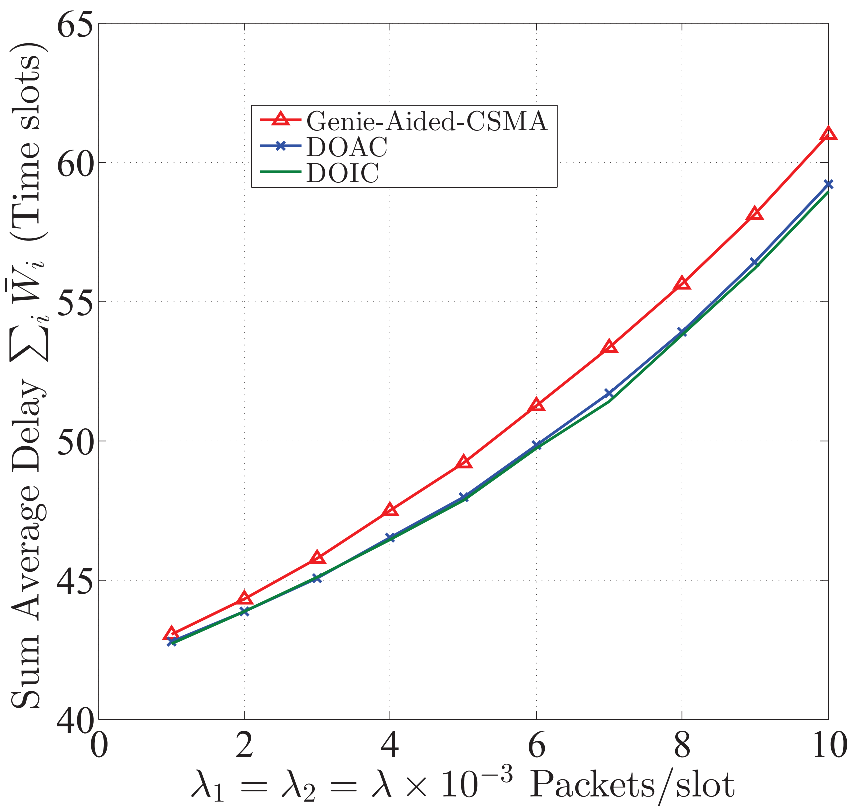

Figure 3.6: Comparing the CSMA policy with the *DOIC* policy and the *DOAC* policy. The power allocation scheme used for the *DOAC* policy is the one used for the CSMA, hence the term genie-aided. However, the genie-aided CSMA policy has a worse delay performance compared to the *DOAC* policy.

3.6.3 Low-Complexity Algorithm Performance

When implementing the suboptimal algorithm proposed in Section 3.3.5 we find that the sum of the average delay across SUs is very close to its optimal value found via Algorithm 2. This is demonstrated in Fig. 3.7 where the error doesn't exceed 0.37% at $\lambda = 0.01$

3.6.4 CSI Estimation Errors

For the imperfect CSI case, we assumed that each SU has an error of $\alpha = 10\%$ in estimating each of $g_i^{(t)}$ and $\gamma_i^{(t)}$ and simulated the system with $d_1 = 32T$. In order

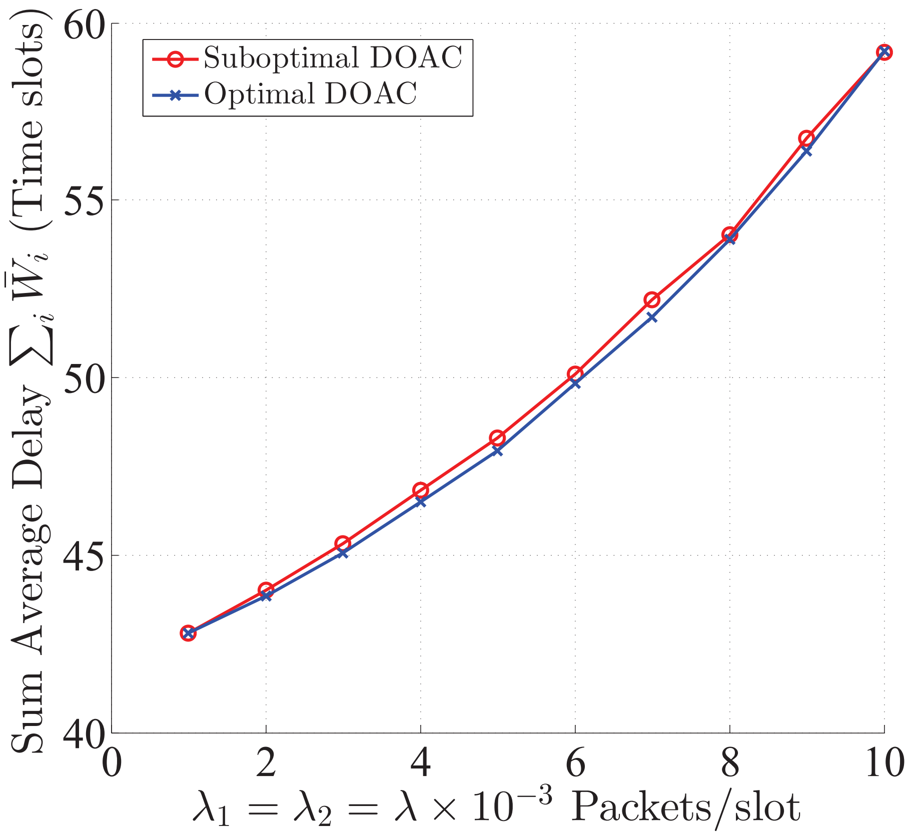

Figure 3.7: The low-complexity algorithm proposed in Section 3.3.5 has a close-to-optimal average delay performance with a maximum error of 0.37%.

to avoid outage we substitute by equation (3.45) in (3.1). To guarantee protection to the PU from interference, we substitute equation (3.46) in (3.20) for the *DOAC* policy, and in (3.15) for the *DOIC* policy. From Fig. 3.8 we see that the performance difference between the perfect and the imperfect CSI problem, for the *DOAC* policy, ranges between 2.4% at $\lambda = 10^{-3}$, and 9.5% at $\lambda = 10^{-2}$. We note that this performance difference represents an upper bound on the actual difference since the 10% is an upper bound on the actual estimation error.

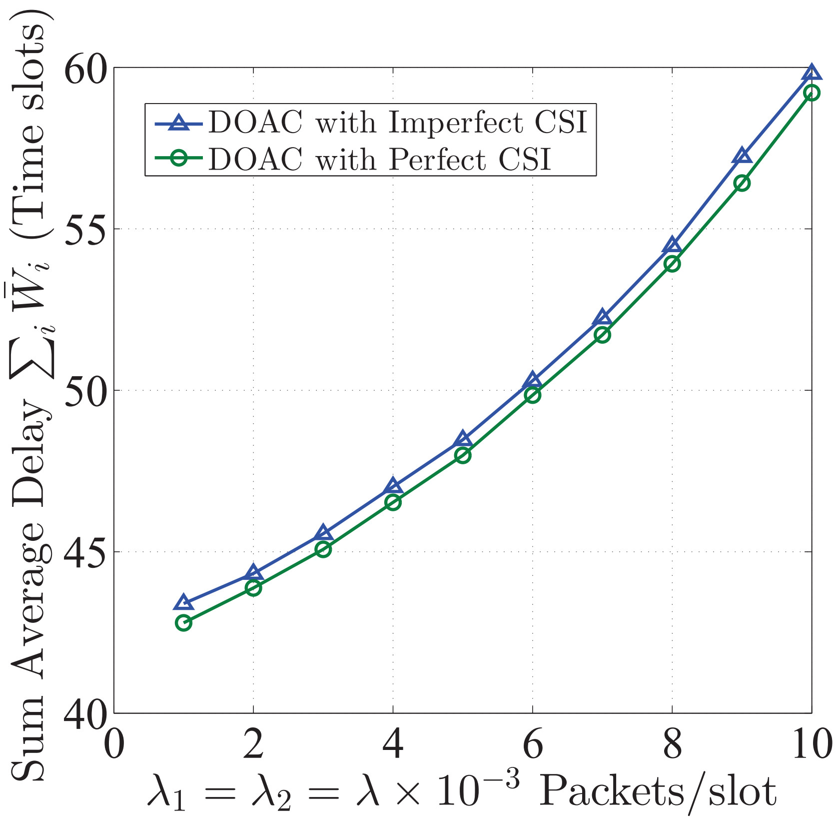

Figure 3.8: Sum of cost functions for the perfect as well as the imperfect channel sensing for the *DOAC* policy to solve the constrained optimization problem (3.8).

Chapter 4

HARD-DEADLINE FRAMEWORK

In the previous chapters, it was assumed that all packets that arrive to the system can be transmitted at any point in time as long as the average delay is bounded. While this assumption might result in an acceptable performance for online streaming of pre-recorded audio/video files, its performance in online streaming of on-air broadcast data such as video conference calls is questionable. This is because, unlike pre-recorded data of a finite time length, video calls have an endless stream of data that needs to arrive in a timely manner. Moreover, not all packets have to be delivered to the end user to have an acceptable QoS for a video call.

Hence, in this chapter another framework for modeling real-time traffic over wireless networks is studied, namely, the “hard-deadline” framework. This model was first introduced in [13] where the authors assume the existence of a deadline associated with each packet that arrives to the system. The packet is considered useful as long as it is transmitted before this deadline and useless, i.e. dropped out of the system and doesn’t count towards the user’s throughput, otherwise. The user is considered a satisfied user if, on average, the percentage of missed packets are below a prespecified threshold. The value of this threshold depends on several factors as the type of application these packets belongs to, the amount of money the user pays and others.

In [13] the authors consider binary erasure channels and present a sufficient and necessary condition to determine if a given problem is feasible. The work is extended in three different directions. The first direction studies the problem under delayed feedback [29]. The second considers general channel fading models [34]. While the third studies multicast video packets that have strict deadlines and utilize network coding to improve the overall network performance [35, 36]. Unlike the time-framed assumption in the previous works, the authors of [11] assume that arrivals and deadlines do not have to occur at the edges of a time frame. They present a scheduling algorithm under the on-off channel fading model and present its achievable region under general arrivals and deadline patterns but with a fixed power transmission. In [37] the authors study the scheduling problem in the presence of real-time and non-real-time data. Unlike real-time data, non-real-time data do not have strict deadlines but have an implicit stability constraint on the queues. Using the dual function approach, the problem was decomposed into an online algorithm that guarantees network stability and real-time users' satisfaction.

Power allocation has not been considered for RT users in the literature, to the best of our knowledge. In this chapter, we study resource allocation in the presence of simultaneous RT and NRT users in a downlink cellular system. We formulate the problem as a joint scheduling-and-power-allocation problem to maximize the sum throughput of the NRT users subject to an average power constraint on the base station (BS), as well as a delivery ratio requirement constraint for each RT user. The delivery ratio constraint requires a minimum ratio of packets to be transmitted by a hard deadline, for each RT user. Perhaps the closest to our work are references [37] and [27]. The former does not consider power allocation, while the latter assumes that only one user can be scheduled per time slot. The contributions in this chapter are as follows:

- We present two scheduling-and-power-allocation algorithms. The first is for the on-off channel fading model while the second is for the continuous channel fading model.
- We show that both algorithms are optimal. That is, both satisfy the aver-

age power constraint, the delivery ratio requirement constraint, in addition to achieving the capacity region. However, the complexity of the first is polynomial in the number of users, while the second is shown to have an average complexity that is close-to-linear.

- We present closed-form expressions for the power allocation policy used by both algorithms. It is shown that the power allocation expressions for the RT and NRT users have a different structure.

- Through simulations, we show the complexity and throughput performances of the proposed algorithms over baseline ones.

- Further, we present a third suboptimal algorithm with linear complexity and compare its throughput performance to the optimal algorithm using simulations.

To provide an outline of how the problem is addressed in this chapter, a summary is provided here. In Section 4.1 we present the system model and the underlying assumptions. The problem is formulated in Section 4.2. For the on-off channel model, the proposed power-allocation and scheduling algorithm as well as its optimality is presented in Section 4.3. In Section 4.4 we present the optimal algorithm for the continuous channel model as well as a low-complexity suboptimal algorithm. The capacity region of the problem is presented in Section 4.5. Simulation results and comparisons with baseline approaches is presented in Section 4.6. Finally, the chapter is concluded in Section 5. I note that the main problem addressed in this chapter could apply for cognitive radio systems as well as regular systems. We present the problem under the latter kind of systems, in this chapter, for ease of presentation.

4.1 System Model

We assume a time slotted downlink system with slot duration T seconds. The system has a single base station (BS) having access to a single frequency channel. There are N users in the system indexed by the set $\mathcal{N} \triangleq \{1, \cdots, N\}$. The set of users is divided into the RT users $\mathcal{N}_{\mathrm{R}} \triangleq \{1, \cdots, N_{\mathrm{R}}\}$, and NRT users $\mathcal{N}_{\mathrm{NR}} \triangleq \{N_{\mathrm{R}} + 1, \cdots, N\}$ with N_{R} and $N_{\mathrm{NR}} \triangleq N - N_{\mathrm{R}}$ denoting the number of RT and NRT users, respectively. Following [13], we model the channel between the BS and the ith user as a fading channel with power gain $\gamma_i(k) = 1$ if it is in a "good" state during the kth slot and $\gamma_i(k) = 0$ otherwise. Channel gains are fixed over the whole slot and change independently in subsequent slots and are independent across users. Hence, the channel gain follows a Bernoulli process. Channels with a more general fading model will be discussed in Section 4.4. Moreover, the channel state information for all users are known to the BS at the beginning of the each slot.

4.1.1 Packet Arrival Model

Let $a_i(k) \in \{0, 1\}$ be the indicator of a packet arrival for user $i \in \mathcal{N}$ at the beginning of the kth slot. $\{a_i(k)\}$ is assumed to be a Bernoulli process with rate λ_i packets per slot and assumed to be independent across all users in the system. Packets arriving at the BS for the RT users are called real-time packets. RT packets have a strict transmission deadline. If an RT packet is not transmitted by this deadline, this packet is dropped out of the system and does not contribute towards the throughput of the user. However, RT user i is satisfied if it receives, on average, more than q_i% of its total number of packets. We refer to this constraint as the QoS constraint for user i. Here we assume that real-time packets arriving at the beginning of the kth slot have their deadline at the end of this slot.

On the other hand, packets arriving to the BS for the NRT users can be

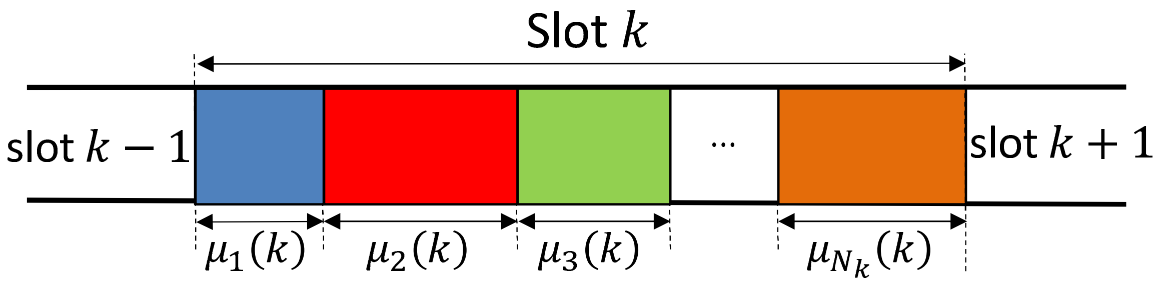

Figure 4.1: In the kth time slot, the BS chooses N_k users to be scheduled. All time slots have a fixed duration of T seconds.

transmitted at any point in time. Thus, packets for NRT user i are stored, at the BS, at user i's (infinite-sized [56]) buffer and served on a first-come-first-serve basis. Since the arrival rate λ_i, for NRT user i, might be higher than what the system can support, we define $r_i(k)$ as an admission controller for user i at slot k. At the beginning of slot k, the BS sets $r_i(k)$ to 1 if the BS decides to admit user i's arrived packet to the buffer, and to 0 otherwise. The time-average number of packets admitted to user i's buffer is

$$A_i \triangleq \limsup_{K\to\infty} \frac{1}{K} \sum_{k=1}^{K} \mathbb{E}\left[r_i(k)\right], \qquad i \in \mathcal{N}_{\mathrm{NR}}. \tag{4.1}$$

And the queue associated with NRT user i is given by

$$Q_i(k+1) = \left(Q_i(k) + Lr_i(k) - \mu_i\left(k\right) R_i(k)\right)^+, \qquad i \in \mathcal{N}_{\mathrm{NR}}, \tag{4.2}$$

where $r_i(k)$ is the admission control decision variable for NRT user i at the beginning of slot k. We note that no admission controller is defined for the RT users since their buffers cannot build up due to the presence of a deadline.

4.1.2 Service Model

Following [34] we assume that more than one user can be scheduled in one time slot. However, due to the existence of a single frequency channel in the system, the BS transmits to the scheduled users sequentially as shown in Fig. 4.1. At the beginning of the kth slot, the BS selects a set of RT users denoted by $\mathcal{S}_{\mathrm{R}}\left(k\right) \subseteq \mathcal{N}_{\mathrm{R}}$ and a set of

NRT users $\mathcal{S}_{\rm NR}(k) \subseteq \mathcal{N}_{\rm NR}$ to be scheduled during slot k. Thus a total of $N_k \triangleq |\mathcal{N}_k|$ users are scheduled at slot k where $\mathcal{N}_k \triangleq \mathcal{S}_{\rm R}(k) \cup \mathcal{S}_{\rm NR}(k)$ (Fig. 4.1). Moreover, the BS assigns an amount of power $P_i(k)$ for every user $i \in \mathcal{N}_k$. This dictates the transmission rate for each user according to the channel capacity given by

$$R_i(k) = \log\left(1 + P_i(k)\,\gamma_i(k)\right). \tag{4.3}$$

Finally, the BS determines the duration of time, out of the T seconds, that will be allocated for each scheduled user. Define the variable $\mu_i(k)$ to represent the duration of time, in seconds, assigned for user $i \in \mathcal{N}$ during the kth slot (Fig. 4.1). Hence, $\mu_i(k) \in [0, T]$ for all $i \in \mathcal{N}$. The BS decides the value of this variable for each user $i \in \mathcal{N}$ at the beginning of slot k. Unlike NRT users which do not have to transmit their packets at a particular time slot, RT users have a strict deadline. Hence, if an RT user was scheduled at slot k, then it should be allocated the channel for a duration of time that allows the transmission of the whole packet. Thus we have

$$\mu_i(k) = \begin{cases} \frac{L}{R_i(k)} & \text{if } i \in \mathcal{S}_{\rm R}(k) \\ 0 & \text{if } i \in \mathcal{N}_{\rm R} \backslash \mathcal{S}_{\rm R}(k) \end{cases}, \tag{4.4}$$

where L is the number of bits per packet, that is assumed to be fixed for all packets in the system. Equation (4.4) means that, depending on the transmission power, if RT user i is scheduled at slot k, then it is assigned as much time as required to transmit its L bits. Hence, unlike for the NRT users where $\mu_i(k) \in [0, T]$, $\mu_i(k)$ is further restricted to the set $\{0, L/R_i(k)\}$ for the RT users. For ease of presentation, we denote $\mathbf{Q}(k) \triangleq [Q_1(k), \cdots, Q_{N_{\rm NR}}(k)]^T$. The BS's goal is solve this power allocation and scheduling problem along with the admission control decisions to maximize the NRT users' sum rate under the system constraints. In the next section we present this problem formally.

4.2 Problem Formulation

We are interested in finding the scheduling and power allocation algorithm that maximizes the sum-rate of all NRT users subject to the system constraints. In this chapter we restrict our search to slot-based algorithms which, by definition, takes the decisions only at the beginning of the slots.

Now define the time-average rate, in packets per slot, of user i to be

$$\overline{R}_i \triangleq \liminf_{K\to\infty} \frac{1}{LTK} \sum_{k=1}^{K} \mu_i\left(k\right) R_i(k), \qquad i \in \mathcal{N}_{\mathrm{NR}} \tag{4.5}$$

while the time-average power consumed by the BS is

$$\bar{P} \triangleq \limsup_{K\to\infty} \frac{1}{K} \sum_{k=1}^{K} P(k) \tag{4.6}$$

where $P(k)$ is the power consumed by the BS during the kth slot which is given by

$$P(k) \triangleq \frac{1}{T} \sum_{i\in\mathcal{N}} P_i\left(k\right) \mu_i\left(k\right). \tag{4.7}$$

Thus the problem we are interested to solve in this work is to find the scheduling, power allocation and packet admission decisions at the beginning of each slot, that solve the following problem

$$\text{maximize} \sum_{i\in\mathcal{N}_{\mathrm{NR}}} \overline{R}_i, \tag{4.8}$$

$$\text{subject to } r_i(k) \leq a_i(k) \qquad \forall i \in \mathcal{N}_{\mathrm{NR}}, \tag{C1}$$

$$\limsup_{k\to\infty} \mathbb{E}\left[Q_i(k)\right] < \infty \qquad \forall i \in \mathcal{N}_{\mathrm{NR}}, \tag{C2}$$

$$\overline{R}_i \geq \lambda_i q_i \qquad \forall i \in \mathcal{N}_{\mathrm{R}}, \tag{C3}$$

$$\bar{P} \leq P_{\mathrm{avg}}, \tag{C4}$$

$$0 \leq P_i\left(k\right) \leq P_{\max} \qquad \forall i \in \mathcal{N}, \tag{C5}$$

$$\sum_{i\in\mathcal{N}} \mu_i\left(k\right) = T \qquad \forall k \geq 1, \tag{C6}$$

$$0 \leq \mu_i\left(k\right) \leq T \qquad \forall i \in \mathcal{N}, \tag{C7}$$

$$\text{variables } \left\{\boldsymbol{\mu}\left(k\right), \mathbf{P}\left(k\right), \boldsymbol{r}\left(k\right)\right\}_{k=1}^{\infty},$$

where $\boldsymbol{\mu}\left(k\right) \triangleq \left[\mu_i(k)\right]_{i\in\mathcal{N}}$, $\mathbf{P}\left(k\right) \triangleq \left[P_i(k)\right]_{i\in\mathcal{N}}$ and $\boldsymbol{r}\left(k\right) \triangleq \left[P_i(k)\right]_{i\in\mathcal{N}_{\mathrm{NR}}}$. Constraint (C1) says that no packets should be admitted to the ith buffer if no packets arrived for user i. Constraint (C2) indicates that the queues of the NRT users are stable when the system reaches steady state. Constraint (C3) indicates that the resources allocated to a RT user i need to be such that the fraction of packets transmitted by the deadline are greater than the required QoS q_i. Constraint (C4) is an average power constraint on the BS transmission power. Finally constraint (C6) guarantees that the sum of durations of transmission of all scheduled users doesn't exceed the slot duration T. In this chapter, we assume that the NRT user with the longest queue has enough packets, at each slot, to fit the whole slot duration which is a valid assumption in the heavy traffic regime. It will be clear that generalizations to the non-heavy traffic regime is possible by allowing multiple NRT users to be scheduled but this is omitted for brevity.

4.3 Proposed algorithm

We use the Lyapunov optimization technique [21] to find and optimal algorithm that solves (4.8). We do this on four steps: i) We define, in Section (4.3.1) a "virtual queue" associated with each average constraint in problem (4.8). This helps in decoupling the problem across time slots. ii) In Section 4.3.2, we define a Lyapunov function, its drift and a, per-slot, reward function. The latter is proportional to the objective of (4.8). iii) Based on the virtual queues and the Lyapunov function, we form an optimization problem, for each slot k, that minimizes the drift-minus-reward expression the solution of which is the proposed power allocation and scheduling algorithm. In Section 4.3.3, we propose an efficient way to solve this problem optimally. iv) Finally, we show that this minimization guarantees reaching an optimal solution for (4.8), in Section 4.3.5.

4.3.1 Problem Decoupling Across Time Slots

We define a virtual queue associated with each RT user as follows

$$Y_i(k+1) = \left(Y_i(k) + a_i(k)q_i - \mathbb{1}_i(k)\right)^+, \qquad i \in \mathcal{N}_{\mathrm{R}}, \tag{4.9}$$

where $\mathbb{1}_i(k) \triangleq \mathbb{1}\left(\mu_i\left(k\right)\right)$ with $\mathbb{1}(\cdot) = 1$ if its argument is non-zero and $\mathbb{1}(\cdot) = 0$ otherwise. For notational convenience we denote $\mathbf{Y}(k) \triangleq [Y_1(k), \cdots, Y_{N_{\mathrm{R}}}(k)]^T$. $Y_i(k)$ is a measure of how much constraint (C3) is violated for user i. We will later show a sufficient condition on $Y_i(k)$ for constraint (C3) to be satisfied. Hence, we say that the virtual queue $Y_i(k)$ is associated with constraint (C3). Similarly, we define the virtual queue $X(k)$, associated with constraint (C4), as

$$X(k+1) = \left(X(k) + \frac{\sum_{i\in\mathcal{N}} P_i\left(k\right)\mu_i\left(k\right)}{T} - P_{\mathrm{avg}}\right)^+. \tag{4.10}$$

To provide a sufficient condition on the virtual queues to satisfy the corresponding constraints, we use the following definition of *mean rate stability* of queues [21, Definition 1] to state the lemma that follows.

Definition 4. *A random sequence* $\{Y_i(k)\}_{k=0}^{\infty}$ *is said to be mean rate stable if and only if* $\limsup_{K\to\infty} \mathbb{E}\left[Y_i(K)\right]/K = 0$ *holds.*

Lemma 7. *If, for some* $i \in \mathcal{N}_{\mathrm{NR}}$*,* $\{Y_i(k)\}_{k=0}^{\infty}$ *is mean rate stable, then constraint* (C3) *is satisfied for user* i*.*

Proof. Proof follows along the lines of Lemma 3 in [21]. □

Lemma 7 shows that when the virtual queue $Y_i(k)$ is mean rate stable, then constraint (C3) is satisfied for user $i \in \mathcal{N}_{\mathrm{NR}}$. Similarly, if $\{X(k)\}_{k=0}^{\infty}$ is mean rate stable, then constraint (C4) is satisfied. Thus, our objective would be to devise an algorithm that guarantees the mean rate stability of $Y_i(k)$ for all RT users as well as the mean rate stability for $X(k)$.

4.3.2 Applying the Lyapunov Optimization

The quadratic Lyapunov function is defined as

$$L_{\mathrm{yap}}\left(U(k)\right) \triangleq \frac{1}{2}\sum_{i\in\mathcal{N}_{\mathrm{R}}} Y_i^2(k) + \frac{1}{2}\sum_{i\in\mathcal{N}_{\mathrm{NR}}} Q_i^2(k) + \frac{1}{2}X^2(k), \tag{4.11}$$

where $\mathbf{U}(k) \triangleq (\mathbf{Y}(k), \mathbf{Q}(k), X(k))$, and the Lyapunov drift as $\Delta(k) \triangleq \mathbb{E}_{U(k)}[L_{k+1}\left(\mathbf{U}(k+1)\right) - L_{\mathrm{yap}}\left(\mathbf{U}(k)\right)]$ where $\mathbb{E}_{\mathbf{U}(k)}\left[x\right] \triangleq \mathbb{E}\left[x|U(k)\right]$ is the conditional expectation of the random variable x given $U(k)$. Squaring (**??**), (4.9) and (4.10) taking the conditional expectation then summing over i, the drift becomes bounded by

$$\Delta(k) \le C_1 + \Psi(k), \tag{4.12}$$

where $C_1 \triangleq \left(\sum_{i\in\mathcal{N}_{\rm R}} \left(q_i^2+1\right) + P_{\max}^2 + P_{\rm avg}^2 + N_{\rm NR}\left[L^2+T^2R_{\max}^2\right]\right)/2$ and we use $R_{\max} \triangleq \log\left(1+P_{\max}\right)$, while

$$\Psi(k) \triangleq \sum_{i\in\mathcal{N}_{\rm R}} \mathbb{E}_{\mathbf{U}(k)}\left[Y_i(k)\left(\lambda_i q_i - \mathbb{1}_i(k)\right)\right] + X(k)\left(\sum_{i\in\mathcal{N}} \frac{\mathbb{E}_{\mathbf{U}(k)}\left[\mu_i\left(k\right) P_i\left(k\right)\right]}{T} - P_{\rm avg}\right)$$
$$+ \sum_{i\in\mathcal{N}_{\rm NR}} Q_i(k)\left(\mathbb{E}_{\mathbf{U}(k)}\left[Lr_i(k) - \mu_i\left(k\right) R_i(k)\right]\right). \tag{4.13}$$

We define $B_{\max}$ as an arbitrarily chosen positive control parameter that controls the performance of the algorithm. We shall discuss the tradeoff on choosing $B_{\max}$ later on. Since $\mathbb{E}_{\mathbf{U}(k)}\left[Lr_i(k)\right]$ represents the average number of bits admitted to NRT user i's buffer at slot k, we refer to $B_{\max}\sum_{i\in\mathcal{N}_{\rm NR}} \mathbb{E}_{\mathbf{U}(k)}\left[Lr_i(k)\right]$ as the "reward term". We subtract this term from both sides of (4.12), then use (4.13) and rearrange to bound the drift-minus-reward term as

$$\Delta(k) - B_{\max}\sum_{i\in\mathcal{N}_{\rm NR}} \mathbb{E}_{\mathbf{U}(k)}\left[Lr_i(k)\right] \leq C_1 + \mathbb{E}_{\mathbf{U}(k)}\left[\sum_{i\in\mathcal{N}_{\rm R}} \Psi_{\rm R}(i,k)\right] + \mathbb{E}_{\mathbf{U}(k)}\left[\sum_{i\in\mathcal{N}_{\rm NR}} \Psi_{\rm NR}(i,k)\mu_i\left(k\right)\right]$$
$$+ \mathbb{E}_{\mathbf{U}(k)}\left[\sum_{i\in\mathcal{N}_{\rm NR}} \left(Q_i(k) - B_{\max}\right) Lr_i(k)\right] + \sum_{i\in\mathcal{N}_{\rm R}} Y_i(k)\lambda_i q_i - X(k)P_{\rm avg}, \tag{4.14}$$

where $\Psi_{\rm R}(i,k)$ and $\Psi_{\rm NR}(i,k)$ are given by

$$\Psi_{\rm R}(i,k) \triangleq \left(Y_i(k) - \frac{L}{TR_i(k)}X(k)P_i\left(k\right)\right)\mathbb{1}_i(k), \qquad i\in\mathcal{N}_{\rm R}, \tag{4.15}$$
$$\Psi_{\rm NR}(i,k) \triangleq Q_i(k)R_i(k) - \frac{X(k)P_i\left(k\right)}{T}, \qquad i\in\mathcal{N}_{\rm NR}, \tag{4.16}$$

respectively, where we used (4.4) in (4.15). The proposed algorithm schedules the users, allocates their powers and controls the packet admission to minimize the right-hand-side of (4.14) at each slot. Since the only term in right-hand-side of (4.14) that is a function in $r_i(k)\ \forall i\in\mathcal{N}_{\rm NR}$ is the fourth term, we can decouple the admission control problem from the joint scheduling-and-power-allocation problem. Minimizing this term results in the following admission controller: set $r_i(k) = a_i(k)$ if $Q_i(k) < B_{\max}$

and 0 otherwise. Minimizing the remaining terms yields

$$\begin{aligned}&\text{maximize}_{\mathbf{P}(k),\boldsymbol{\mu}(k)} \textstyle\sum_{i\in\mathcal{S}_{\mathrm{R}}(k)} \Psi_{\mathrm{R}}(i,k) + \sum_{i\in\mathcal{N}_{\mathrm{NR}}} \Psi_{\mathrm{NR}}(i,k)\mu_i\,(k)\\ &\text{subject to (C5), (C6) and (C7).}\end{aligned} \tag{4.17}$$

This is a per-slot optimization problem the solution of which is an algorithm that minimizes the upper bound on the drift-minus-reward term defined in (4.14). Next we show how to solve this problem in an efficient way.

4.3.3 Efficient Solution for the Per-Slot Problem

We first solve for the NRT variables then use its result to solve for the RT variables.

4.3.3.1 NRT variables

To solve this problem optimally, we first find the optimal power-allocation-and-scheduling policy for the NRT users through the following lemma.

Lemma 8. *If an NRT user i is scheduled to transmit any of its NRT data during the kth slot, then the optimum power level for this NRT with respect to (w.r.t.) problem* (4.17) *is given by*

$$P_i\,(k) = \min\left(\left(\frac{Q_i(k)}{X(k)} - 1\right)^+, P_{\max}\right). \tag{4.18}$$

Moreover, in the heavy traffic regime, the optimum NRT user to be scheduled, if any, w.r.t. problem (4.17) *is* $i^*_{\mathrm{NR}} \triangleq \arg\max_{i\in\mathcal{N}_{\mathrm{NR}}} Q_i(k)$.

Proof. We observe that, for any $i \in \mathcal{N}_{\mathrm{NR}}$, the only term in (4.17) that is a function in $P_i\,(k)$ is $\Psi_{\mathrm{NR}}(i,k)$. Differentiating (4.16) w.r.t. $P_i\,(k)$ for all $i \in \mathcal{N}_{\mathrm{NR}}$, equating the results to 0 and noting the minimum and maximum power constraints (C5), we get the water-filling power allocation formula (4.18). This completes the first part of the lemma.

To prove the second part, we substitute by (4.18) in (4.16) to get

$$\Psi^*_{\mathrm{NR}}(i,k) \triangleq Q_i(k)\log\left(Q_i(k)\right) - Q_i(k) + X(k) - Q_i(k)\log\left(X(k)\right), \tag{4.19}$$

then we note that it is easy to show that $\arg\max_{i\in\mathcal{N}_{\mathrm{NR}}} Q_i(k) = \arg\max_{i\in\mathcal{N}_{\mathrm{NR}}} \Psi^*_{\mathrm{NR}}(i,k)$. We continue the proof by contradiction. Suppose that the optimal scheduled NRT set is given by $\mathcal{S}^*_{\mathrm{NR}}(k) = \{i^*_{\mathrm{NR}}, j\}$ where $j \neq i^*_{\mathrm{NR}}$ and $\Psi^*_{\mathrm{NR}}(j,k) < \Psi^*_{\mathrm{NR}}(i^*_{\mathrm{NR}},k)$. Thus, there exists some values $\alpha > 0$ and $\beta > 0$ such that the corresponding scheduler would be $\mu_{i^*_{\mathrm{NR}}}(k) = \alpha$ and $\mu_j(k) = \beta$, while $\mu_{l_k}(k) = 0$ for all $l_k \notin \{i^*_{\mathrm{NR}}, j\}$. In other words, α seconds are assigned to i^*_{NR} and β seconds assigned to j. However, if user i^*_{NR} has enough backlogged data, which happens in the heavy traffic regime, then we can increase its assigned duration to $\mu_{i^*_{\mathrm{NR}}} = \alpha + \beta$ and thus set $\mu_j(k) = 0$, to get an increase in the objective of (4.17) by $\beta\left(\Psi^*_{\mathrm{NR}}(i^*_{\mathrm{NR}},k) - \Psi^*_{\mathrm{NR}}(j,k)\right) > 0$ which contradicts with the optimality of $\mathcal{S}^*_{\mathrm{NR}}(k)$ and completes the proof of the lemma. $\square$

Lemma 8 provides the optimal scheduling policy for the NRT users, at the kth slot, as well as the optimal power allocation w.r.t. problem (4.17). The lemma shows that if any of the NRT users is going to be scheduled in the kth slot, then only one of them is going to be scheduled. This means that the scheduling policy for the NRT users is

$$\mu_i(k) = \begin{cases} T - \sum_{i\in\mathcal{S}^*_{\mathrm{R}}(k)} \mu_i(k) & i = i^*_{\mathrm{NR}} \\ 0 & \mathcal{N}_{\mathrm{NR}} \backslash \{i^*_{\mathrm{NR}}\} \end{cases} \tag{4.20}$$

which is a manipulation of (C6). Substituting (4.20) and (4.19) in (4.17), the latter becomes

$$\begin{aligned} \underset{\mu_{i^*_{\mathrm{NR}}}(k),[\mu_i(k),P_i(k)]_{i\in\mathcal{N}_{\mathrm{NR}}}}{\text{maximize}} \quad & \sum_{i\in\mathcal{S}_{\mathrm{R}}(k)} \Psi_{\mathrm{R}}(i,k) + \Psi^*_{\mathrm{NR}}(i^*_{\mathrm{NR}},k)\mu_{i^*_{\mathrm{NR}}}(k) \qquad (4.21)\\ \text{subject to} \quad & (\mathrm{C7}), (\mathrm{C5}) \text{ and } \mu_{i^*_{\mathrm{NR}}}(k) = T - \sum_{i\in\mathcal{S}_{\mathrm{R}}(k)} \frac{L}{\log\left(1 + P_i(k)\,\gamma_i(k)\right)}, \end{aligned}$$

which is simpler than (4.17) since it is not a function in the NRT variables except $\mu_{i^*_{\mathrm{NR}}}(k)$. Finding the optimal value of $\mu_{i^*_{\mathrm{NR}}}(k)$ solves the NRT scheduling problem. We will first solve for $\mu_i(k)$ for all RT users then use (4.20) to find $\mu_{i^*_{\mathrm{NR}}}(k)$.

4.3.3.2 RT Variables

To find the scheduler of the RT users that is optimal w.r.t. problem (4.21), we first solve for $[P_i(k)]_{i\in\mathcal{N}_{\mathrm{R}}}$ given a fixed set $\mathcal{S}_{\mathrm{R}}(k)$, then we discuss the scheduling policy that solves for this set. To solve for $[P_i(k)]_{i\in\mathcal{N}_{\mathrm{R}}}$, we present the following definition then present a theorem that discusses the optimum power allocation policy for the RT users.

Definition 5. *We define the Lambert power allocation policy for the RT users as*

$$P_i(k) = \min\left(\frac{\frac{T\Psi^*_{\mathrm{NR}}(i^*_{\mathrm{NR}},k)}{X(k)} - 1}{W_0\left(\left[\frac{\Psi^*_{\mathrm{NR}}(i^*_{\mathrm{NR}},k)T}{X(k)} - 1\right]e^{-1}\right)} - 1, P_{\max}\right), \qquad i \in \mathcal{S}_{\mathrm{R}}(k), \quad (4.22)$$

where $W_0(z)$ *is the principle branch of the Lambert W function [47] while* $\Psi^*_{\mathrm{NR}}(i,k)$ *is given in* (4.19)*.*

Theorem 5. *Given any set* $\mathcal{S}_{\mathrm{R}}(k)$*, if the Lambert power policy results in* $\sum_{i\in\mathcal{S}_{\mathrm{R}}(k)} L/\log(1+P_i(k)) \leq T$*, then it is the optimum RT-users' power allocation policy given that* $\mathcal{S}_{\mathrm{R}}(k)$ *is the scheduling set at slot* k*. Otherwise, the optimum power allocation policy is given by*

$$P_i(k) = e^{\frac{|\mathcal{S}_{\mathrm{R}}(k)|L}{T}} - 1, \qquad i \in \mathcal{S}_{\mathrm{R}}(k). \quad (4.23)$$

Proof. We prove this theorem by applying the Lagrange optimization [45, Ch. 5] technique to problem (4.21) then use the complementary slackness condition.

Since $\mu_i(k) \geq 0$ for all $i \in \mathcal{N}_{\mathrm{R}}$ (see (4.4)), then we have the constraint $\mu_{i^*_{\mathrm{NR}}}(k) \leq T$ always holds from (4.20). Thus we define the Lagrange multiplier ϕ to be the multiplier associated with the constraint $\mu_{i^*_{\mathrm{NR}}}(k) \geq 0$. The Lagrangian becomes

$$L_{\text{agr}} \triangleq \sum_{i\in\mathcal{S}_{\text{R}}(k)} \Psi_{\text{R}}(i,k) + \left(\Psi^{*}_{\text{NR}}(i^{*}_{\text{NR}},k)+\phi\right)\left(T - \sum_{i\in\mathcal{S}_{\text{R}}(k)} \frac{L}{\log\left(1+P_i\left(k\right)\gamma_i(k)\right)}\right) \tag{4.24}$$

Differentiating (4.24) with respect to $P_i\left(k\right)$ and equating to 0 gives

$$-\frac{\log\left(1+P_i\left(k\right)\gamma_i(k)\right)\frac{X(k)L}{T} - \frac{\left(X(k)P_i(k)/T+\Psi^{*}_{\text{NR}}(i^{*}_{\text{NR}},k)+\phi\right)\gamma_i(k)}{1+P_i(k)\gamma_i(k)}}{\log^2\left(1+P_i\left(k\right)\gamma_i(k)\right)} = 0. \tag{4.25}$$

After some manipulations and denoting $\tilde{\phi} \triangleq \left(\Psi^{*}_{\text{NR}}(i^{*}_{\text{NR}},k)+\phi\right)T/X(k)$ we get

$$\log\left(1+P_i\left(k\right)\gamma_i(k)\right) = 1 + \frac{\tilde{\phi}\gamma_i(k)-1}{1+P_i\left(k\right)\gamma_i(k)} \triangleq 1+\tilde{P}. \tag{4.26}$$

Thus we get $\tilde{P}e^{\tilde{P}} = \left(\tilde{\phi}\gamma_i(k)-1\right)e^{-1}$ which has two solutions in $\tilde{P}$ (see [47]), one of them yields a negative value for $P_i\left(k\right)$. Hence, with the help of $W_0(\cdot)$, which is the inverse function of xe^x, we can write a unique solution for (4.25) as

$$P_i\left(k\right) = \frac{1}{\gamma_i(k)}\left[\frac{\tilde{\phi}\gamma_i(k)-1}{W_0\left(\left[\tilde{\phi}\gamma_i(k)-1\right]e^{-1}\right)} - 1\right], \qquad i\in\mathcal{S}_{\text{R}}\left(k\right). \tag{4.27}$$

To calculate (4.27), we need to find the value of ϕ satisfying the complementary slackness condition $\phi\mu_{i^{*}_{\text{NR}}}\left(k\right)=0$. Hence we have one of the two following possibilities might yield the optimal solution: 1) setting $\phi=0$ and thus $\mu_{i^{*}_{\text{NR}}}\left(k\right)\geq 0$, or 2) setting $\mu_{i^{*}_{\text{NR}}}\left(k\right)=0$ and thus $\phi\geq 0$. If setting $\phi=0$ yields $\sum_{i\in\mathcal{S}_{\text{R}}(k)} L/\log(1+P_i\left(k\right))\leq T$ then the Lambert power allocation policy in (4.22) is optimum since there exists no other non-negative value for ϕ that yields $\sum_{i\in\mathcal{S}_{\text{R}}(k)} L/\log(1+P_i\left(k\right))=T$ while satisfying $\mu_{i^{*}_{\text{NR}}}\left(k\right)=0$ (to satisfy the complementary slackness). On the other hand, if setting $\phi=0$ yields $\sum_{i\in\mathcal{S}_{\text{R}}(k)} L/\log(1+P_i\left(k\right))>T$, then ϕ cannot be 0. Thus we have $\mu_{i^{*}_{\text{NR}}}\left(k\right)=0$, which means that the time slot will be allocated for RT users only. The corresponding value of ϕ should satisfy $\sum_{i\in\mathcal{S}_{\text{R}}(k)} L/\log(1+P_i\left(k\right))=T$. From (4.27), we observe that $P_i\left(k\right)=P_j(k)$ for all $i,j\in\mathcal{S}_{\text{R}}\left(k\right)$ because $\gamma_i(k)=1$ for all $i \in \mathcal{S}_{\mathrm{R}}(k)$. Thus we have $L|\mathcal{S}_{\mathrm{R}}(k)|/\log(1+P_i(k)) = T$. This yields the power allocation policy (4.23) and completes the proof. $\square$

4.3.4 Algorithms

In this subsection, we present two algorithms to solve problem (4.17) optimally. The first algorithm has a linear complexity while the second one makes use of the structure of the problem to reduce the complexity even below linear. Although both algorithms are equivalent, the latter is the one used in the MATLAB simulations.

4.3.4.1 Linear-Complexity Algorithm

Theorem 5 gives closed-form expressions for the power function of the RT users given any scheduling set $\mathcal{S}_{\mathrm{R}}(k)$. To find the optimum scheduling set $\mathcal{S}_{\mathrm{R}}(k)$ that solves problem (4.21), we present the following definition then mention a theorem that decreases the complexity of this search.

Definition 6. *At slot k, the set $\mathcal{S}_{\mathrm{R}}(k)$ is said to be a "candidate" set if and only if $Y_i(k) \geq Y_j(k)$ for all $i \in \mathcal{S}_{\mathrm{R}}(k)$ and all $j \notin \mathcal{S}_{\mathrm{R}}(k)$. Otherwise it is called a "non-candidate" set.*

We note that the definition of candidate sets assumes that all RT users have $\gamma_i(k) = 1$. If this assumption does not hold at some time slot k, then we eliminate the users with $\gamma_i(k) = 0$ from the system for this time slot and consider only those with $\gamma_i(k) = 1$.

Theorem 6. *The optimal RT set that solves* (4.21) *is one of the candidate sets.*

Proof. We prove this theorem by contradiction. Suppose that $\mathcal{S}^*_{\mathrm{R}}(k)$ is the optimal set and that it is not a candidate set. That is, $\exists i \in \mathcal{S}_{\mathrm{R}}(k)$ and $j \notin \mathcal{S}_{\mathrm{R}}(k)$ such that $Y_i(k) < Y_j(k)$. It is easy to show that the Lambert power policy results in the fact that $P_i(k)$ depends on $|\mathcal{S}_{\mathrm{R}}(k)|$ and not on $\mathcal{S}_{\mathrm{R}}(k)$ for any $i \in \mathcal{S}_{\mathrm{R}}(k)$ and any $\mathcal{S}_{\mathrm{R}}(k)$. Thus, replacing user i with user j results in having $P_j(k) = P_i(k)$ which means that $X(k)P_j(k)\mu_j(k) = X(k)P_i(k)\mu_i(k)$ holds. But since $Y_i(k) < Y_j(k)$, swapping the two users increases the objective function of (4.21) and results in a candidate set. This contradicts with the fact that $\mathcal{S}_{\mathrm{R}}(k)$ is optimal while being non-candidate. □

Theorem 6 says that there will be no scheduled RT users having a value of $Y_j(k)$ smaller than any of the unscheduled RT users. This theorem suggests an algorithm to reduce the complexity of scheduling the RT users from $O\left(2^{N_{\mathrm{R}}}\right)$ to $O(N_{\mathrm{R}})$. This algorithm is to list the RT users in a descending order of their $Y_i(k)$. Without loss of generality, in the remaining of this paper, we will assume that $Y_1 > Y_2 \cdots > Y_{N_{\mathrm{R}}}$.

We now propose Algorithm 3 which is the scheduling and power allocation algorithm for problem (4.8). Algorithm 3 is executed at the beginning of the kth slot and, without loss of generality, it assumes: 1) all RT users in the system have received a packet at the beginning of the kth slot, 2) all users in the system have an "on" channel. If, at some slot, any of these assumptions does not hold for some users, these users are eliminated from the system for this slot. That is, they will not be scheduled. In addition, we assume heavy traffic regime, thus the NRT user with the longest queue has enough data to fill the entire time slot. We define the set $\mathcal{S}_{\mathrm{RT}}$ to be the set of all candidate sets.

4.3.4.2 Simpler Algorithm used in MATLAB Simulations

Under the Lambert power policy let's define l as the number of RT users can be scheduled in slot k under the Lambert policy. l is given by

$$l \triangleq \left\lfloor \frac{T}{\mu_i(k)} \right\rfloor \tag{4.28}$$

Algorithm 3 Scheduling and Power Allocation Algorithm

Define the auxiliary functions $\Psi_{\mathrm{X}}(\cdot) : \mathcal{S}_{\mathrm{RT}} \rightarrow \mathbb{R}_+$ and $P_{\mathrm{X}}(\cdot,\cdot) : \mathcal{S}_{\mathrm{RT}} \times \mathcal{N}_{\mathrm{R}} \rightarrow \mathbb{R}_+$.
Initialize $P_{\mathrm{X}}(\mathcal{S}, i) = 0$ for all $\mathcal{S} \in \mathcal{S}_{\mathrm{RT}}$ and all $i \in \mathcal{N}_{\mathrm{R}}$.
Sort the RT users in a descending order of $Y_i(k)$. Without loss of generality, assume that $Y_1 > Y_2 \cdots > Y_{N_{\mathrm{R}}}$.
Find the user i^*_{NR} with longest queue $Q_i(k)$ and set $\mathcal{S}_{\mathrm{R}}(k)$ to be an empty set.
while $i \leq N_{\mathrm{R}}$ **do**
  $\mathcal{S}_{\mathrm{R}}(k) = \mathcal{S}_{\mathrm{R}}(k) \cup \{i\}$ and set the power according to (4.22) $\forall i \in \mathcal{S}_{\mathrm{R}}(k)$.
  Calculate $\mu_i(k)$ and $\mu_{i^*_{\mathrm{NR}}}(k)$ according to (4.4) and (4.20), respectively.
  if $\mu_{i^*_{\mathrm{NR}}}(k) < 0$ **then**
    Set $\mu_i(k) = 0$ for all $i \in \mathcal{N}_{\mathrm{NR}}$ and set the power allocation for all $i \in \mathcal{S}_{\mathrm{R}}(k)$ according to (4.23) and recalculate $\mu_i(k)$ according to (4.4).
  end if
  Set $\Psi_{\mathrm{X}}(\mathcal{S}_{\mathrm{R}}(k)) = \sum_{i \in \mathcal{S}_{\mathrm{R}}(k)} (Y_i(k) - X_i(k)\mu_i(k)) + \Psi^*_{\mathrm{NR}}(i^*_{\mathrm{NR}}, k)\mu_{i^*_{\mathrm{NR}}}(k)$.
  Set $P_{\mathrm{X}}(\mathcal{S}_{\mathrm{R}}(k), i) = P_i(k)$, $\forall i \in \mathcal{S}_{\mathrm{R}}(k)$.
  $i \leftarrow i + 1$.
end while
Set the optimum scheduling set $\mathcal{S}^*_{\mathrm{R}}(k) = \arg\max_{\mathcal{S}_{\mathrm{R}}(k)} \Psi_{\mathrm{X}}(\mathcal{S}_{\mathrm{R}}(k))$.
Set $P^*_i(k) = P_{\mathrm{X}}(\mathcal{S}^*_{\mathrm{R}}(k), i)$ for all $i \in \mathcal{N}_{\mathrm{R}}$, and set the NRT scheduler according to (4.20).
For each $i \in \mathcal{N}_{\mathrm{NR}}$, set $r_i(k) = a_i(k)$ if $Q_i(k) < B_{\max}$ and 0 otherwise.
Update equations (**??**), (4.9) and (4.10) at the end of the kth slot.

Before presenting the algorithm that solves problem (4.21) and the theorem behind it we present the following two conditions on l that will facilitate the understanding of the algorithm and the presentation of the theorem.

Condition 1. $l = 0$.

Condition 2. $0 < l \leq N_{\mathrm{R}}$ *and the following two inequalities hold*

$$Y_l(k) > \left[X(k)P_l(k) + \Psi^*_{\mathrm{NR}}(i^*_{\mathrm{NR}}, k)\right]\mu_l(k) \tag{4.29}$$

$$Y_{l+1}(k) \geq X(k)\left[T\left(e^{\frac{(l+1)L}{T}} - 1\right) - P_l(k)l\mu_l(k)\right] + \Psi^*_{\mathrm{NR}}(i^*_{\mathrm{NR}}, k)\left(T - l\mu_l(k)\right) \tag{4.30}$$

Condition 1 means that the duration $\mu_i(k)$ of one RT user is greater than the slot duration under the Lambert policy. On the other hand, Condition 2 means that, roughly speaking, the $Y_i(k)$ values are very high to the extent that the RT users are suffering more than the NRT users during slot k. The next theorem shows that when any of Conditions 1 or 2 holds, the BS should schedule only RT users during slot k.

Theorem 7. *To solve* (4.21)*, if either Condition 1 or Condition 2 holds, then the optimal scheduling for the RT users is given by*

$$\mathcal{S}_{\mathrm{R}}(k) = \left\{ i : Y_i > X(k)T\left(e^{\frac{iL}{T}} - e^{\frac{(i-1)L}{T}}\right)\right\}. \tag{4.31}$$

On the other hand if neither of these conditions holds, then the optimal scheduling policy for the RT users is

$$\mathcal{S}_{\mathrm{R}}(k) = \{i : Y_i > [X(k)P_i(k) + \Psi^*_{\mathrm{NR}}(i^*_{\mathrm{NR}}, k)]\,\mu_i(k)\} \tag{4.32}$$

while the optimum NRT user is i^*_{NR} with its duration calculated from (4.20)*.*

Proof. To find the optimum set $\mathcal{S}_{\mathrm{R}}(k)$, we assume, for simplicity of presentation, that $Y_1 > Y_2 > \cdots > Y_{N_{\mathrm{R}}}$. Given two sets $\mathcal{S}_{\mathrm{R}}(k) \subset \mathcal{N}_{\mathrm{R}}$ and $\mathcal{S}'_{\mathrm{R}}(k) \triangleq \mathcal{S}_{\mathrm{R}}(k) \cup \{j\}$ for some $j \in \mathcal{N}_{\mathrm{R}} \backslash \mathcal{S}_{\mathrm{R}}(k)$, if we find that the inequality $\sum_{i\in\mathcal{S}_{\mathrm{R}}(k)} L/\log(1 + P_i(k)) > T$ holds under the Lambert power policy, then it is easy to show that inequality $\sum_{i\in\mathcal{S}'_{\mathrm{R}}(k)} L/\log(1 + P_i(k)) > T$ still holds, under the Lambert power policy. We note that inequality $\sum_{i\in\mathcal{S}_{\mathrm{R}}(k)} L/\log(1 + P_i(k)) > T$ means that the sum of durations of the RT users exceeds the slot duration which indicates that no NRT users can be scheduled. This means that, according to Theorem 5, if no NRT users are going to be scheduled given some set $\mathcal{S}_{\mathrm{R}}(k)$, then adding more users to this set would not result in scheduling any NRT users. Similarly, we can show that if NRT user i^*_{NR} is scheduled under set $\mathcal{S}_{\mathrm{R}}(k)$, then removing any users from this set would not let this NRT user become not scheduled. □

We now propose Algorithm 4 which is the scheduling and power allocation algorithm for problem (4.8). Algorithm 4 is executed at the beginning of the kth slot and, without loss of generality, it assumes: 1) all RT users in the system have received a packet at the beginning of the kth slot and 2) all users in the system have an "on" channel. If, at some slot, any of these assumptions does not hold for some users, these users are eliminated from the system for this slot. That is, they will not be scheduled.

Algorithm 4 MATLAB Simulation Algorithm

1: Sort the RT users in a descending order of $Y_i(k)$. Without loss of generality, assume that $Y_1 > Y_2 \cdots > Y_{N_{\mathrm{R}}}$.
2: Find the user i^*_{NR} with longest queue $Q_i(k)$.
3: Set the power according to (4.22) for all RT users.
4: For the RT users calculate $\mu_i(k)$ and l according to (4.4) and (4.28), respectively.
5: **if** Condition 1 OR Condition 2 holds **then**
6: Set $\mu_i(k) = 0$ for all $i \in \mathcal{N}_{\mathrm{NR}}$ and set the scheduling and power allocation of the RT users according to (4.31) and (4.23), respectively.
7: **else**
8: Schedule the RT users according to

$$\mu_i(k) = \begin{cases} 1 & i \in \mathcal{S}_{\mathrm{R}}(k) \\ 0 & \text{otherwise} \end{cases} \tag{4.33}$$

where $\mathcal{S}_{\mathrm{R}}(k)$ is given in (4.32) and set the RT users' powers according to (4.22).
9: Schedule the NRT users according to (4.20) and set user i^*_{NR}'s power $P_{i^*_{\mathrm{NR}}}(k)$ via (4.18).
10: **end if**
11: For each $i \in \mathcal{N}_{\mathrm{NR}}$, set $r_i(k) = a_i(k)$ if $Q_i(k) < B_{\max}$ and 0 otherwise.
12: Update equations (4.2), (4.9) and (4.10) at the end of the kth slot.

4.3.5 Optimality of Proposed Algorithm

We first define $R_i^{(\mathrm{opt})}$ to be the throughput of NRT user i under the optimal algorithm that solves (4.8). We define this algorithm to be the one that sets, at each time slot k, the variables $P_i(k)$, $\mu_i(k)$, $\mathbb{1}_i(k)$ and $R_i(k)$ to the values $\tilde{P}_i(k)$, $\tilde{\mu}_i(k)$, $\tilde{\mathbb{1}}_i(k)$ and

$\tilde{R}_i(k)$, respectively, where the latter values satisfy

$$\begin{aligned} \limsup_{K\to\infty} \tfrac{1}{K}\textstyle\sum_{k=0}^{K-1} \mathbb{E}\left[\tilde{\mathbb{1}}_i(k)\right] \geq & \quad \lambda_i q_i, \quad \forall i \in \mathcal{N}_{\mathrm{R}}, & (4.34)\\ \limsup_{K\to\infty} \tfrac{1}{K}\textstyle\sum_{k=0}^{K-1}\sum_{i\in\mathcal{N}} \mathbb{E}\left[\frac{\tilde{\mu}_i(k)\tilde{P}_i(k)}{T}\right] \leq & \quad P_{\mathrm{avg}}, & (4.35)\\ \limsup_{K\to\infty} \tfrac{1}{K}\textstyle\sum_{k=0}^{K-1} \mathbb{E}\left[\frac{\tilde{\mu}_i(k)\tilde{R}_i(k)}{L}\right] = & \quad R_i^{(\mathrm{opt})}, \quad \forall i \in \mathcal{N}_{\mathrm{NR}}, & (4.36) \end{aligned}$$

where $R_i^{(\mathrm{opt})}$ is the optimal rate for user $i \in \mathcal{N}_{\mathrm{NR}}$ with respect to solving (4.8). The following theorem gives a bound on the performance of Algorithm 3 compared to the optimal algorithm that has a genie-aided knowledge of $R_i^{(\mathrm{opt})}$ which, we show that, due to this knowledge it can solve the problem optimally.

Theorem 8. *If $\gamma_i(k) \in \{0,1\}$ for all $i \in \mathcal{N}$ and all $k \geq 1$, then for any $B_{\max} > 0$ Algorithm 3 results in satisfying all constraints in* (4.8) *and achieves an average rate satisfying*

$$\sum_{i\in\mathcal{N}_{\mathrm{NR}}} \overline{R}_i \geq \sum_{i\in\mathcal{N}_{\mathrm{NR}}} R_i^{(\mathrm{opt})} - \frac{C_1}{LB_{\max}}. \tag{4.37}$$

Proof. See Appendix G □

Theorem 8 says that Algorithm 3 yields an objective function (4.8) that is arbitrary close to the performance of the optimal algorithm that solves (4.8).

4.4 Continuous Fading

In the case of continuous fading, i.e. $\gamma_i(k) \in [0, \gamma_{\max}]$ where $\gamma_{\max} < \infty$ is the maximum channel gain that $\gamma_i(k)$ can take, we expect the power allocation to depend on the channel gain. An algorithm that solves this case is a generalization of Algorithm 3 that assumes $\gamma_i(k) \in \{0,1\}$. However, as will be demonstrated later, the scheduling algorithm of the RT users has a higher complexity order than the special case of on-off channel gains.

4.4.1 System Model and Problem Formulation

We adopt the same model as in Section 4.1 except that we allow $\gamma_i(k)$ to take any value in the interval $[0, \gamma_{\max}]$, for all $i \in \mathcal{N}$. The transmission rate for this case is still given by (4.3), and the optimization problem is the same as (4.8) with the new definition of $\gamma_i(k)$.

4.4.2 Derivation of Algorithm

Algorithm 5 is based on the same Lyapunov optimization procedure as in Section 4.3.2. Following this procedure, we reach optimization problem (4.17).

Lemma 9. *If user $i \in \mathcal{N}_{\mathrm{NR}}$ is scheduled to transmit any of its NRT data during the kth slot, then the optimum power level for this NRT w.r.t. problem* (4.17) *in the continuous fading case is given by*

$$P_i(k) = \min\left(\left(\frac{Q_i(k)}{X(k)} - \frac{1}{\gamma_i(k)}\right)^+, P_{\max}\right). \tag{4.38}$$

Moreover, in the heavy traffic regime, the scheduled NRT user, if any, that optimally solves problem (4.8) *is given by*

$$i^*_{\mathrm{NR}} = \arg\max_{i \in \mathcal{N}_{\mathrm{NR}}} \Psi^*_{\mathrm{NR}}(i, k), \tag{4.39}$$

with ties broken randomly uniformly, while

$$\Psi^*_{\mathrm{NR}}(i,k) \triangleq Q_i(k)\log\left(Q_i(k)\right) - Q_i(k) + \frac{X(k)}{\gamma_i(k)} - Q_i(k)\log\left(\frac{X(k)}{\gamma_i(k)}\right). \tag{4.40}$$

Proof. The proof is similar to that of Lemma 8 and is omitted for brevity. □

Lemma 9 presents the optimal power and scheduling policy for the NRT users. To solve for the RT users, we assume a fixed subset $\mathcal{S}_{\mathrm{R}}(k) \subseteq \mathcal{N}_{\mathrm{R}}$ of RT users to be scheduled during the kth slot and find the power allocation of these users. Consequently, the optimum set $\mathcal{S}_{\rm R}^*(k)$ is the one that maximizes (4.17). We present this algorithm and its optimality in Section 4.4.3. Motivated by the high complexity of this algorithm, we present, in Section 4.4.4, a heuristic to schedule the RT users and show its performance to the optimal one by simulations.

Assuming that the users in the set $\mathcal{S}_{\rm R}(k)$ are scheduled at the kth slot, the problem is to find the transmission power levels for all the users in this set. We answer this question in the following theorem.

Theorem 9. *In the continuous-fading channel model, given some non-empty set* $\mathcal{S}_{\rm R}(k)$*, the power allocation policy*

$$P_i(k) = \min\left(\frac{1}{\gamma_i(k)}\left[\frac{\tilde{\phi}\gamma_i(k)-1}{W_0\left(\left[\tilde{\phi}\gamma_i(k)-1\right]e^{-1}\right)}-1\right], P_{\max}\right), \qquad i \in \mathcal{S}_{\rm R}(k), \tag{4.41}$$

with $\tilde{\phi} \triangleq (\Psi_{\rm NR}^*(i_{\rm NR}^*, k)+\phi)\, T/X(k)$ *and* $\Psi_{\rm NR}^*(i_{\rm NR}^*, k)$ *given by* (4.40)*, is optimal w.r.t.* (4.17) *when* ϕ *is set to a non-negative value that satisfies* (C6)*. Moreover, an upper-bound on the lagrange multiplier* ϕ *can be given by*

$$\phi \le \phi_{\max} \triangleq -\Psi_{\rm NR}^*(i_{\rm NR}^*, k) + \frac{e^{\frac{L|\mathcal{S}_{\rm R}(k)|}{T}} L|\mathcal{S}_{\rm R}(k)|X(k)P_{\max}}{e^{\frac{L|\mathcal{S}_{\rm R}(k)|}{T}}-1} \tag{4.42}$$

Proof Sketch: The proof is similar to that of Theorem 5. The only difference is that we have to obtain the optimum value of ϕ satisfying (C6). We note that instead of finding $\phi > 0$ using a 1-dimensional grid search, we can use the bisection method [48, Ch.9] which requires the monotonicity of the left-hand-side of (C6), a fact that can be shown easily by showing that the derivative, of this left-hand-side, with respect to ϕ is always negative. Moreover, since the bisection algorithm needs a bracketing interval, it can be easily shown that the optimum ϕ satisfies $\phi \le \phi_{\max}$. □

It is clear that the Lambert power policy in (4.41) has a totally different structure than the water-filling policy in (4.38). The reason is because the former is for the RT users while the later is for the NRT users. We plot the two policies in Fig. 4.2 with $L = 1$, $T = 1$, $P_{\max} = 20$ while $Q_i(k)/X(k) = 15$. The Lambert policy is plotted assuming a single RT user is scheduled at slot k while the water-filling policy is plotted assuming a single NRT user is scheduled at slot k. We note that when a RT user i is the only scheduled user, (4.41) is equivalent to

$$P_i\left(k\right) = \min\left(\frac{e^{L/T} - 1}{\gamma_i(k)}, P_{\max}\right), \tag{4.43}$$

We demonstrate that, while the water-filling is an increasing function in the channel gain, the Lambert is a decreasing function in the channel gain. This is because the RT user has a single packet of a fixed length to be transmitted. If the channel gain increases, then the power decreases to keep the same transmission rate resulting in the same transmission duration of one slot. This result holds when multiple RT users are scheduled as well as demonstrated in the following theorem.

Theorem 10. *Let $\mathcal{N}_k$ be some scheduling RT set for* (4.17) *at slot k. For a fixed set $\mathcal{N}_k$, the power $P_i\left(k\right)$ given by* (4.41) *is monotonically decreasing in $\gamma_i(k)$ $\forall i \in \mathcal{S}_{\mathrm{R}}\left(k\right) \subseteq \mathcal{N}_k$.*

Proof Sketch: Proof follows by differentiating (4.41) with respect to $\gamma_i(k)$ for some user i, while having ϕ satisfying (C6), and showing that the resulting derivative is always non-positive for $\gamma_i(k) \geq 0$. $\square$

The optimum scheduling algorithm for the RT users is to find, among all subsets of the set $\mathcal{N}_{\mathrm{R}}$, the set that gives the highest objective function of (4.17).

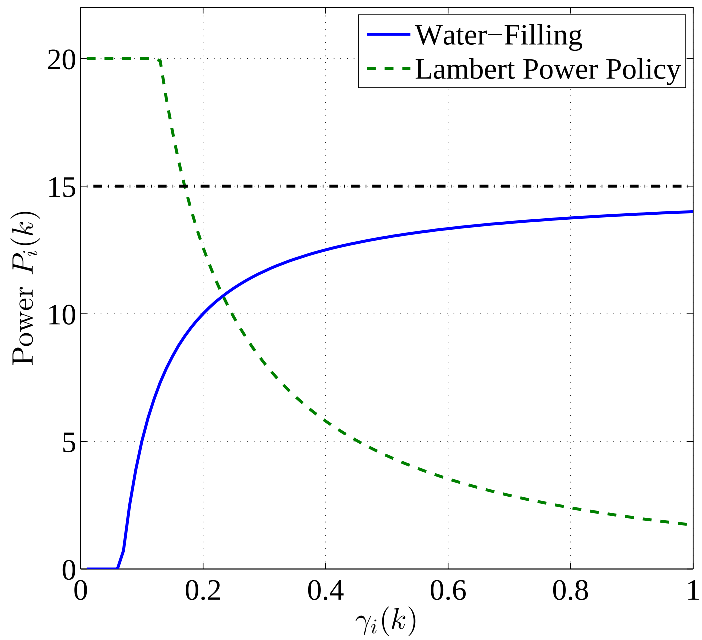

Figure 4.2: The Lambert power policy decreases with the channel gain, while the water-filling policy increases with the gain.

4.4.3 Proposed Algorithm and Proof of Optimality

The exhaustive approach to the scheduling problem is to evaluate the objective function of (4.21) for all $2^{N_{\mathrm{R}}}$ possible sets and choose the set that gives the highest objective function. This may be not practical when the number of RT users is large. Observing the approach in the special on-off case and inspired by Theorem 6 that reduces the search space, we provide here a similar approach. We first provide the following definition which is analogous to Theorem 6.

Theorem 11. *At slot k, for any set $\mathcal{S}_{\mathrm{R}}(k)$, if there exists some $i \notin \mathcal{S}_{\mathrm{R}}(k)$ and some $j \in \mathcal{S}_{\mathrm{R}}(k)$ such that $Y_i(k) > Y_j(k)$ and $\gamma_i(k) > \gamma_j(k)$, then $\mathcal{S}_{\mathrm{R}}(k)$ cannot be an optimal RT set, with respect to problem (4.21), for the continuous channel model.*

Proof Sketch: The proof is carried out by contradiction. We can show that if $Y_i(k) > Y_j(k)$ and $\gamma_i(k) > \gamma_j(k)$ for some $i \notin \mathcal{S}_{\mathrm{R}}(k)$ and some $j \in \mathcal{S}_{\mathrm{R}}(k)$, then we could form another set $\mathcal{S}'(k)$ by swapping users i and j and thus increase the objective function of (4.21). □

This theorem provides a sufficient condition for non-optimality. In other words, we can make use of this theorem to restrict our search algorithm to the sets that do not satisfy this property. Before presenting the proposed algorithm, we define the set $\mathcal{S}_{\mathrm{RT}}$ as the set of all possible subsets of the set $\mathcal{N}_{\mathrm{R}}$.

Theorem 12. *If $\gamma_i(k) \in [0, \gamma_{\max}]$ for all $i \in \mathcal{N}$ and all $k \geq 1$, then for any $B_{\max} > 0$ and any $\epsilon \in (0, 1]$ there exists some finite constant C_2 such that Algorithm 5 satisfies all constraints in (4.8) and achieves an average sum throughput satisfying*

$$\sum_{i \in \mathcal{N}_{\mathrm{NR}}} \overline{R}_i \geq \sum_{i \in \mathcal{N}_{\mathrm{NR}}} \overline{R}_i^* - \frac{C_2}{L B_{\max}}, \tag{4.47}$$

where $\overline{R}_i^*$ is the optimal rate for user i w.r.t. (4.8).

Proof. The proof is similar to that of Theorem 8 and C_2 is defined as C_1 but with $R_{\max} \triangleq \log(1 + P_{\max}\gamma_{\max})$. We omit the proof for brevity. □

Due to the problem being a combinatorial problem with a huge amount of possibilities, we could not reach a closed-form expression for the complexity order of this algorithm. However, simulations will show its complexity improvement over the exhaustive search algorithm.

4.4.4 Heuristic

In some time slots, where the condition in step 5 of Algorithm 5 is not satisfied for all sets $\mathcal{S} \in \mathcal{S}_{\mathrm{RT}}$, the complexity is $2^{N_{\mathrm{R}}}$. This could be not practical when the number of RT users is large. Observing the approach in the special on-off case and motivated

Algorithm 5 Lambert-Strict Algorithm

1: Define the auxiliary functions $\Psi_{\mathrm{X}}(\cdot) : \mathcal{S}_{\mathrm{RT}} \to \mathbb{R}_+$ and $P_{\mathrm{X}}(\cdot, \cdot) : \mathcal{S}_{\mathrm{RT}} \times \mathcal{N}_{\mathrm{R}} \to \mathbb{R}_+$.
2: Initialize $P_{\mathrm{X}}(\mathcal{S}, i) = 0$ for all $\mathcal{S} \in \mathcal{S}_{\mathrm{RT}}$ and all $i \in \mathcal{N}_{\mathrm{R}}$.
3: Find the user i^*_{NR} given in (4.39) and calculate its power given by (4.38).
4: **for** $\mathcal{S} \in \mathcal{S}_{\mathrm{RT}}$ **do**
5: **if** $\exists$ some $i \notin \mathcal{S}$ and some $j \in \mathcal{S}$ such that $Y_i(k) > Y_j(k)$ and $\gamma_i(k) > \gamma_j(k)$ **then**
6: Set $\Psi_{\mathrm{X}}(\mathcal{S}) = -\infty$.
7: Skip this iteration and go to step 4 to continue with the next set in $\mathcal{S}_{\mathrm{RT}}$.
8: **end if**
9: $\phi \leftarrow \phi_{\max} + \Delta\phi$
10: **while** $\phi \mu_i(k) \neq 0$ **do**
11: $\phi \leftarrow \phi - \Delta\phi$
12: Calculate $P_i(k)$ given by (4.41) for all $i \in \mathcal{S}$ and set $\mu_{i^*_{\mathrm{NR}}}(k) = T - \sum_{i \in \mathcal{S}} \mu_i(k)$.

$$\mu_{i^*_{\mathrm{NR}}}(k) = T - \sum_{i \in \mathcal{S}} \mu_i(k) \tag{4.44}$$

13: **end while**
14: Set

$$\Psi_{\mathrm{X}}(\mathcal{S}) = \sum_{i \in \mathcal{S}} \left(Y_i(k) - X_i(k)\mu_i(k)\right) + \Psi^*_{\mathrm{NR}}(i^*_{\mathrm{NR}}, k)\mu_{i^*_{\mathrm{NR}}}(k), \tag{4.45}$$

$$\text{and } P_{\mathrm{X}}(\mathcal{S}, i) = P_i(k), \qquad i \in \mathcal{S}. \tag{4.46}$$

15: $i \leftarrow i + 1$.
16: **end for**
17: Set the optimum scheduling set $\mathcal{S}^*_{\mathrm{R}}(k) = \arg\max_{\mathcal{S}} \Psi_{\mathrm{X}}(\mathcal{S})$.
18: Set $P^*_i(k) = P_{\mathrm{X}}\left(\mathcal{S}^*_{\mathrm{R}}(k), i\right)$ for all $i \in \mathcal{N}_{\mathrm{R}}$, and set the NRT scheduler according to (4.20).
19: For each $i \in \mathcal{N}_{\mathrm{NR}}$, set $r_i(k) = a_i(k)$ if $Q_i(k) < B_{\max}$ and 0 otherwise.
20: Update equations (4.2), (4.9) and (4.10) at the end of the kth slot.

by Lemma 6, we find that RT users who have a higher value for $Y_i(k)$ tend to be scheduled more since higher $Y_i(k)$ indicates more violation for the QoS constraint (C3). However, having a higher $Y_i(k)$ is not a sufficient condition for scheduling user i. That is, we might have $Y_i(k) > Y_j(k)$ for some $i, j \in \mathcal{N}_\mathrm{R}$ but the optimum scheduler schedules user j but not user i. This could be the case only if user j gives a higher objective function in problem (4.17). This happens only if the inequality $P_j(k)\mu_j(k)$ is sufficiently smaller than $P_i\left(k\right)\mu_i(k)$ (see (4.17)).

These two arguments are contradicting; on one hand we should favor users who have higher $Y_i(k)$ values, while on the other we should favor those who have a lower $P_i\left(k\right)\mu_i(k)$. Hence, the heuristic proposed in Algorithm 6 applies the first argument in the odd time slots and the second argument in the even ones. In other words, in the odd time slots it assigns higher scheduling priorities to users with higher $Y_i(k)$ over those with lower ones. While in the even time slots, users with a lower value of $P_i\left(k\right)\mu_i(k)$ have higher scheduling priorities.

Before presenting the proposed algorithm, we note that sorting the users in a descending order of $\gamma_i(k)$ is equivalent to sorting them in an ascending order of the quantity $P_i\left(k\right)\mu_i(k)$. This result decreases the complexity since we do not have to calculate the quantity $P_i\left(k\right)\mu_i(k)$ which needs the calculation of the Lambert W function as well as the value of ϕ.

Algorithm 6 has a similar structure as Algorithm 5. The difference is in sorting the users where the former sorts them according to $Y_i(k)$ and $\gamma_i(k)$ in the odd and even time slots, respectively, while the latter does not sort them. Moreover, the search for the scheduling set of the RT users is of $O(N_\mathrm{R})$ as compared to the exponential complexity of $O(2^{N_\mathrm{R}})$ of Algorithm 5.

Algorithm 6 Scheduling and Power Allocation Algorithm

if k is odd **then**
  Sort the RT users in a descending order of $Y_i(k)$. Without loss of generality, assume that $Y_1(k) > Y_2(k) \cdots > Y_{N_{\mathrm{R}}}(k)$.
else
  Sort the RT users in a descending order of $\gamma_i(k)$. Without loss of generality, assume that $\gamma_1(k) > \gamma_2(k) \cdots > \gamma_{N_{\mathrm{R}}}(k)$.
end if
Define the auxiliary functions $\Psi_{\mathrm{X}}(\cdot) : \mathcal{S}_{\mathrm{RT}} \rightarrow \mathbb{R}_+$ and $P_{\mathrm{X}}(\cdot, \cdot) : \mathcal{S}_{\mathrm{RT}} \times \mathcal{N}_{\mathrm{R}} \rightarrow \mathbb{R}_+$.
Initialize $P_{\mathrm{X}}(\mathcal{S}, i) = 0$ for all $\mathcal{S} \in \mathcal{S}_{\mathrm{RT}}$ and all $i \in \mathcal{N}_{\mathrm{R}}$.
Find the user i^*_{NR} given in (4.39) and calculate its power given by (4.38).
Set $i = 1$ and $\mathcal{S} = \{\}$.
while $i \leq N_{\mathrm{R}}$ **do**
  Set $\mathcal{S} = \mathcal{S} \cup \{i\}$ and $\phi \leftarrow \phi_{\max} + \Delta\phi$
  while $\phi\mu_i(k) \neq 0$ **do**
    $\phi \leftarrow \phi - \Delta\phi$
    Calculate $P_i(k)$ given by (4.41) for all $i \in \mathcal{S}$ and set $\mu_{i^*_{\mathrm{NR}}}(k)$ according to (4.44).
  end while
  Set $\Psi_{\mathrm{X}}(\mathcal{S})$ and $P_{\mathrm{X}}(\mathcal{S}, i)$ according to (4.46) and (4.45), respectively.
end while
Set the optimum scheduling set $\mathcal{S}^*_{\mathrm{R}}(k) = \arg\max_{\mathcal{S}} \Psi_{\mathrm{X}}(\mathcal{S})$.
Set $P^*_i(k) = P_{\mathrm{X}}(\mathcal{S}^*_{\mathrm{R}}(k), i)$ for all $i \in \mathcal{N}_{\mathrm{R}}$, and set the NRT scheduler according to (4.20).

4.5 Capacity Region

In Section 4.4, Algorithm 5 is shown to maximize the NRT sum-throughput subject to the system constraints. In this section we want to study the stability of the system. Specifically, we are interested to answer the following two questions:

1. What is the capacity region of the system under the continuous fading model?

2. What scheduling and power-allocation algorithms can achieve this capacity region?

Studying the system's capacity region means that we need to find all arrival rate vectors $\boldsymbol{\lambda}_{\mathrm{NR}}$ under which the NRT users' queues are stable (i.e. have a stationary distribution). This needs to be studied assuming that all arriving packets are admitted to their respective buffers. Hence we first eliminate the admission controller $\boldsymbol{r}(k)$ by replacing the queue equation (4.2) with

$$Q_i(k+1) = \left(Q_i(k) + La_i(k) - \mu_i(k)\, R_i(k)\right)^+ . \tag{4.48}$$

More formally, the first question now becomes: what is the closure of all admissible arrival rate vectors? An admissible arrival rate vector is defined next.

Definition 7. *An arrival rate vector $\boldsymbol{\lambda}_{\mathrm{NR}}$ is said to be admissible if there exists a power-allocation and scheduling algorithm under which constraints* (C3) *and* (C2) *are satisfied given the power and scheduling constraints* (C4), (C5), (C6) *and* (C7).

For simplicity we henceforth assume that the channel gain $\gamma_i(k) \in \mathcal{M}$ where $\mathcal{M}$ is a discrete finite set, the elements of which are in the range $[0, \gamma_{\max}]$. With a slight abuse in notation, we define $\gamma_i(\boldsymbol{m}, k) \triangleq \gamma_i(k)$ to be the gain of user i when the channel is in fading state $\boldsymbol{m} \triangleq [\gamma_1(\boldsymbol{m}), \cdots, \gamma_N(\boldsymbol{m})]^T \in \mathcal{M}^N$ during slot k. We also define $\mu_i(\boldsymbol{m}, k)$ and $P_i(\boldsymbol{m}, k)$ to be, respectively, the duration and power allocated to user $i \in \mathcal{N}$ when the channel is in fading state $\boldsymbol{m} \triangleq [\gamma_1(\boldsymbol{m}), \cdots, \gamma_N(\boldsymbol{m})]^T \in \mathcal{M}^N$ during slot k, and $\pi_{\mathbf{m}}$ to be the probability of occurrence of fading state $\boldsymbol{m}$. We now mention the following definition then state Theorem 13 that answers the first question.

Definition 8. *An arrival rate vector $\boldsymbol{\lambda}_{\mathrm{NR}}$ is said to belong to the "Lambert Region" $\mathcal{R}_{\mathrm{Lamb}}$ if and only if there exists a sequence of time duration vectors $\{\boldsymbol{\mu}(\boldsymbol{m}, k)\}$ and a power allocation policy $\{\mathbf{P}(\boldsymbol{m}, k)\}$ that make $\boldsymbol{\lambda}_{\mathrm{NR}}$ satisfy*

$$\lambda_i = \frac{1}{L} \sum_{\boldsymbol{m} \in \mathcal{M}^N} \mu_i(\boldsymbol{m}, k) \log\left(1 + P_i(\boldsymbol{m}, k)\, \gamma_i(\boldsymbol{m}, k)\right) \pi_{\mathbf{m}}, \qquad i \in \mathcal{N}_{\mathrm{NR}}, \tag{4.49}$$

while having $\{\boldsymbol{\mu}(k)\}$ *and* $\{\mathbf{P}(k)\}$ *satisfy*

$$q_i\lambda_i \leq \quad \textstyle\sum_{\boldsymbol{m}\in\mathcal{M}^N} \mu_i(\boldsymbol{m},k)\log\left(1+P_i(\boldsymbol{m},k)\,\gamma_i(\boldsymbol{m},k)\right), \qquad i\in\mathcal{N}_{\mathrm{R}}, \tag{4.50}$$

$$\textstyle\sum_{i\in\mathcal{N}} \mu_i(\boldsymbol{m},k) \leq T, \qquad \forall k\geq 1, \boldsymbol{m}\in\mathcal{M}^N, \tag{4.51}$$

$$\limsup_{K\to\infty}\tfrac{1}{K}\textstyle\sum_{k=1}^{K}\sum_{i\in\mathcal{N}}\sum_{m\in\mathcal{M}}\mu_i(\boldsymbol{m},k)\,P_i(\boldsymbol{m},k)\leq P_{\mathrm{avg}}, \tag{4.52}$$

$$\mu_i(\boldsymbol{m},k)\geq 0, \qquad i\in\mathcal{N}, \forall k\geq 1, \boldsymbol{m}\in\mathcal{M}^N, \tag{4.53}$$

$$P_i(\boldsymbol{m},k)\geq 0, \qquad i\in\mathcal{N}, \forall k\geq 1, \boldsymbol{m}\in\mathcal{M}^N. \tag{4.54}$$

Theorem 13. *If* $\boldsymbol{\lambda}_{\mathrm{NR}}(1+\epsilon)\in\mathcal{R}_{\mathrm{Lamb}}$ *then Algorithm 5 satisfies* (C2)-(C7)*. Otherwise, then problem* (4.8) *is infeasible.*

Proof. See Appendix H □

Theorem 13 says that $\mathcal{R}_{\mathrm{Lamb}}$ is in fact the system's capacity region. This answers the first question. Moreover, the second question is answered in the proof, as shown in Appendix H. In the proof, we show that with a simple modification to Algorithm 5 we can achieve this capacity region. The modification is by setting $r_i(k)=a_i(k)$ for all $i\in\mathcal{N}_{\mathrm{NR}}$.

4.6 Simulation Results

We simulate the system for the on-off channel model as well as the continuous channel model. For both models, we assume that all channels are statistically homogeneous, i.e. $\bar{\gamma}_i = \bar{\gamma}$ for all $i\in\mathcal{N}$ where $\bar{\gamma}$ is a fixed constant. Moreover, all RT users have homogeneous delivery ratio requirements, thus $q_i=q$ for all $i\in\mathcal{N}_{\mathrm{R}}$ for some parameter q. All parameter values are summarized in Table 4.1 for all simulation figures unless otherwise specified.

We compare the throughput of the RT users, which is the objective of problem (4.8), to that of a simple power allocation and scheduling algorithm that we call

Table 4.1: Simulation Parameter Values

Parameter	Value	Parameter	Value
L	1 bit/packet	$P_{\max}$	200
$B_{\max}$	10^4	$\bar{\gamma}_i$, $\forall i$	1
$\{q_i\}_{i \in \mathcal{N}_{\mathrm{R}}}$	0.3	T	1

"FixedP" algorithm. In the FixedP algorithm, all scheduled users transmit with the maximum power, i.e. $P_i(k) = P_{\max}$ for all $i \in \mathcal{N}$ and all $k \geq 1$, while the scheduling policy is to flip a biased coin and choose to schedule either the NRT users or the RT users. The coin is set to schedule the RT users with probability q (the delivery ratio requirement for all users), at which case the RT users are sorted according to $Y_i(k)$ and scheduled one by one until the current slot ends. On the other hand, when the coin chooses the NRT users, the FixedP policy assigns the entire time slot to the NRT user with the longest queue.

4.6.0.1 On-Off Channel Model

We assume that we have $N = 20$ users that is split equally between the RT and NRT users, i.e. $N_{\mathrm{R}} = N_{\mathrm{NR}} = 20$. Fig. 4.3 shows a substantial increase in the average rate of the proposed algorithm over the FixedP algorithm with over 200% at low P_{avg} values and 60% at high P_{avg} values.

In Fig. 4.4, the sum of average NRT users' throughput is plotted while keeping $P_{\mathrm{avg}} = 10$ but changing q. We can see that the FixedP algorithm results in a large degradation in the throughput compared to Algorithm 3 which allocates the power and schedules the users optimally with respect to (4.8). The decrease in the throughput observed in both curves of Fig. 4.4 is due to the increase in the parameter q. This increase makes constraint C3 more stringent and thus decreases the feasible region decreasing the throughput.

In Fig. 4.5 we show the effect of increasing the number of users on the system's

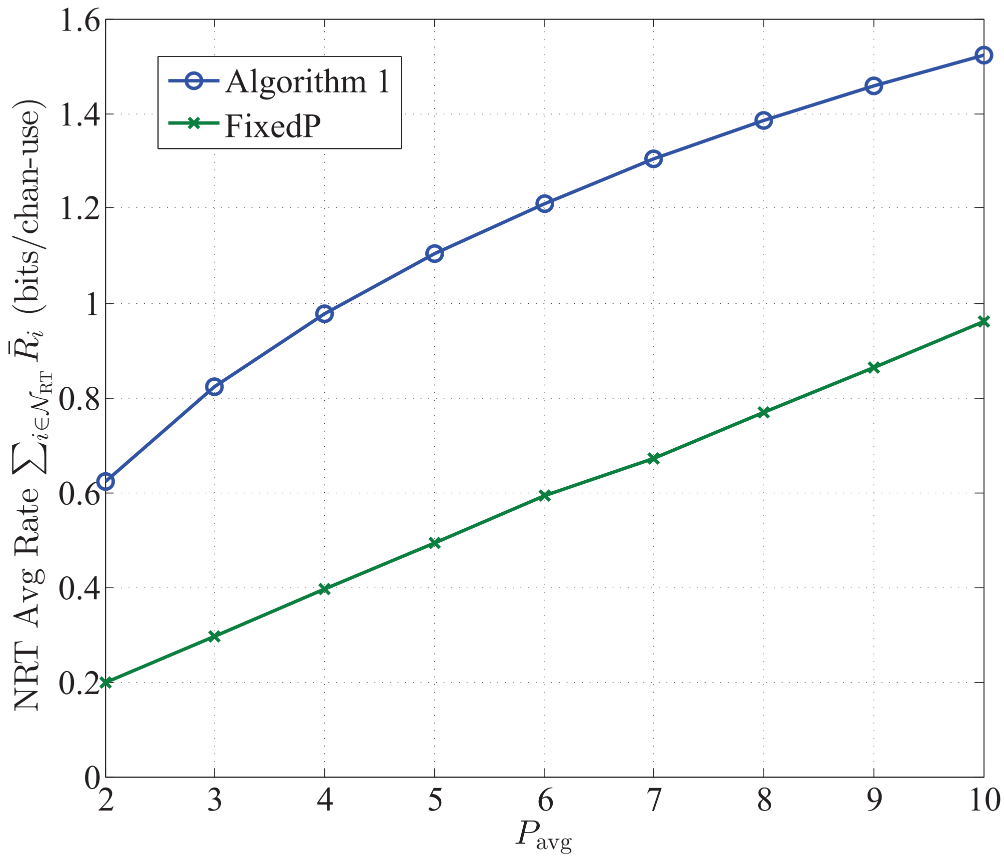

Figure 4.3: Sum of average throughput for all NRT users. The FixedP algorithm assigns a fixed power to all users set at P_{max}.

throughput. As the number of users increase, more RT users have to be scheduled. This comes at the expense of the time allocated to the NRT users thus decreasing the throughput for the two plotted algorithms.

4.6.0.2 Continuous Channel Fading Model

In this simulation setup, we assume the channels are fading according to a Rayleigh fading model with avg power gain of $\bar{\gamma} = 1$.

In Fig. 4.6, we plot the average rate performance of the optimal (Algorithm 5) as well as the suboptimal (Algorithm 6) algorithms versus the number of users $N_{\mathrm{NR}} = N_{\mathrm{R}}$. As the number of users increase more RT users are scheduled, on average,

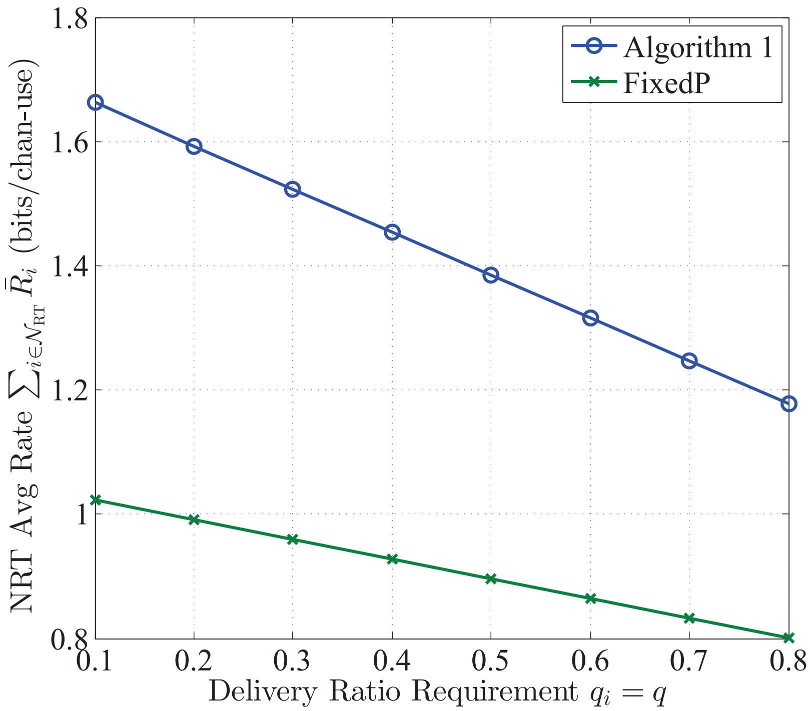

Figure 4.4: As q increases, the RT users are assigned the channel more frequently. This comes at the expense of the NRT's throughput. However, the proposed algorithm outperforms the FixedP algorithm.

per time slot. This comes at the expense of the NRT users' throughput. Although the number of NRT users increase as well creating multi-user diversity effect, we do not observe an increase in the throughput. This is because NRT users are not scheduled based on the channel gain only but on the queue length as well.

In Fig. 4.7, we plot the complexity of the Lambert-Strict algorithm as well as the exhaustive search algorithm with exponential complexity versus the number of users N_{R}. The complexity is measured in terms of the average number of iterations, per-slot, where we have to evaluate the objective function of (4.21). Since this complexity changes from a slot to the other, we plot the average of this complexity. As the number of users increases, the Lambert-Strict algorithm has an average complexity close to linear.

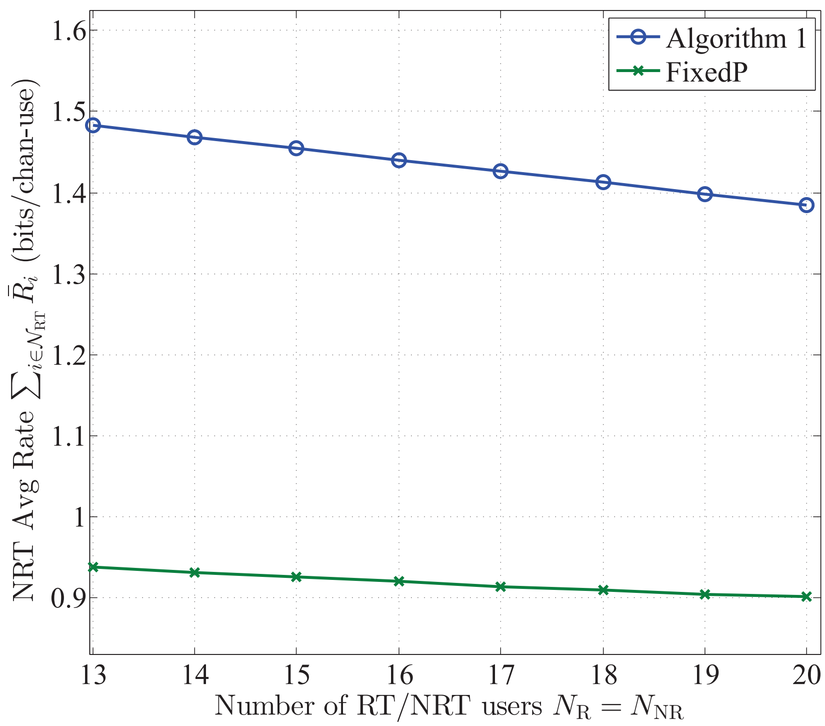

Figure 4.5: As N increases, the RT users are allocated the channel more at the expense of the NRT users' throughput.

4.7 Conclusions

We discussed the problem of throughput maximization in downlink cellular systems in the presence of RT and NRT users. We formulated the problem as a joint power-allocation-and-scheduling problem. Using the Lyapunov optimization theory, we presented two algorithms to optimally solve the throughput maximization problem. The first algorithm is optimal under the on-off channel fading model, while the second is for the continuous channel fading model. The power allocations for both algorithms are in closed-form expressions for the RT as well as the NRT users. We showed that the NRT power allocation is water-filling-like which is monotonically increasing in the channel gain. On the other hand, the RT power allocation has a totally different structure that we call the "Lambert Power Allocation". It is found that the latter is

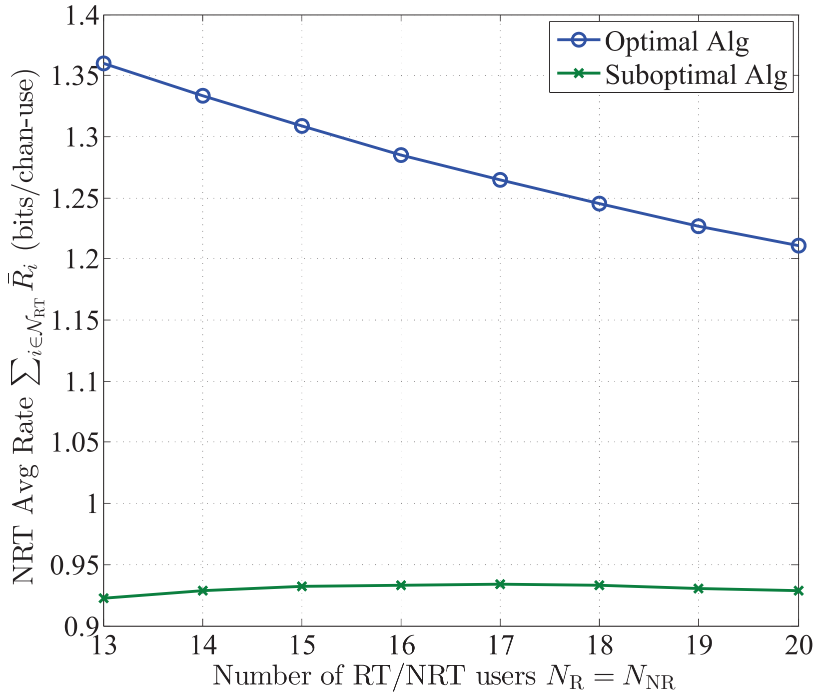

Figure 4.6: As the number of NRT users in the system increase the throughput of the RT users decrease since this constitutes more load to the system.

a decreasing function in the channel gain.

The two algorithms differ in the complexity of the adopted scheduling policies. The first algorithm that assumes on-off channel model has a polynomial-time complexity. While the second algorithm that works for on-off as well as continuous channel models has an exponential scheduling complexity. Motivated by this large complexity, we present a heuristic algorithm that is shown, through simulations, to give at least as half as the throughput of the optimal algorithm. We presented the capacity region of the problem and showed that the proposed algorithms achieve the capacity region.

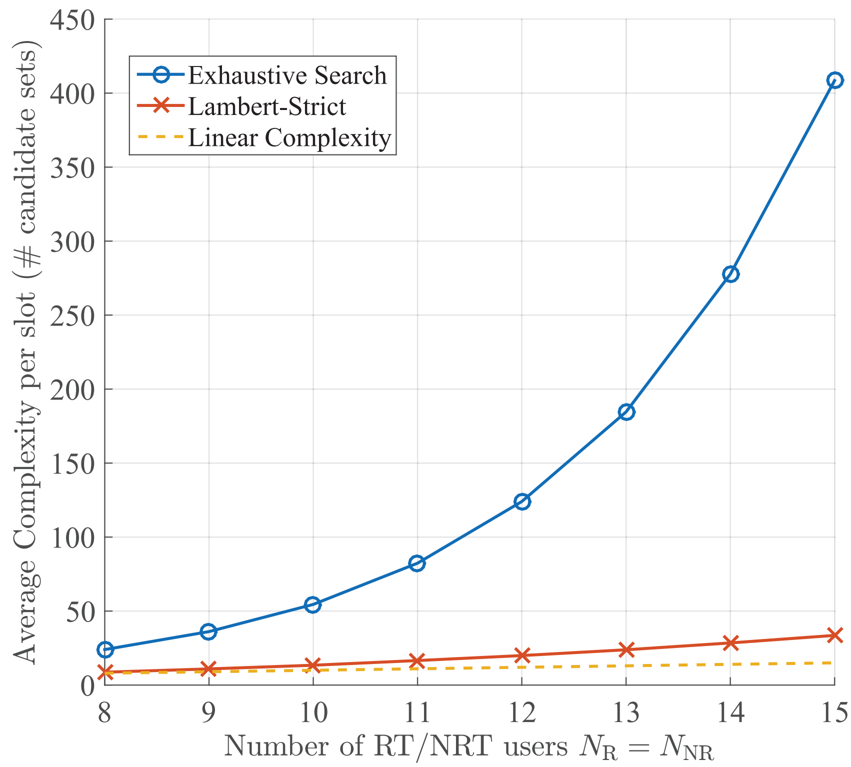

Figure 4.7: As the number of NRT users in the system increase the complexity increases exponentially for Algorithm 5 and nearly linear for the Lambert-Strict one.

Chapter 5

DISSERTATION CONCLUSIONS

In this dissertation I have studied the joint scheduling and power allocation problem of a cellular CR system. The main goal of this work is to present low-complexity joint power allocation and scheduling algorithms that provide acceptable quality of service (QoS) for the real-time packets of the SUs while protecting the PU. I have studied two well-known frameworks, namely, the average-delay framework where a constrained is imposed on the average delay of each packet, and the hard-deadline framework where a constraint is imposed on the instantaneous delay of each packet.

Since the service time is a common factor in both frameworks, I studied the delay due to the service time in a single SU multichannel system first where the SU senses the channels sequentially and stops to transmit at the first channel that gives the highest throughput. The goal was to find the optimal power that maximizes the SU's throughput as well as guaranteeing that the SU's service time is below some prespecified threshold. I formulated the problem as an optimal stopping rule problem and presented a closed-form expression for the stopping rule. The algorithm was proven to guarantee that the PU is protected from harmful interference and its performance was compared to numerous baseline algorithms.

Then, I studied the problem under the average-delay framework. The problem was formulated as a delay minimization problem in the presence of average and instantaneous interference constraints to the PU, as well as an average delay constraint for each SU that needs to be met. Most of the existing literature that studies this problem either assume on-off fading channels or does not provide a delay-optimal algorithms which is essential for real-time applications. I proposed the *DOAC* policy that dynamically updates the power allocation of the SUs as well as finding the optimal scheduling policy. The scheduling policy is found by dynamically updating a priority list based on the statistics as well as the history of the arrivals, departures and channel fading realizations. The proposed algorithm updates the priority list on a per-frame basis while controlling the power on a per-slot basis. I showed, through the Lyapunov optimization, that the proposed $DOAC$ policy is asymptotically delay optimal. That is, it minimizes the sum of any convex increasing function of the average delays of the SUs as well as satisfying the average interference and average delay constraints. However, it is found that when the number of SUs N in the system is large, the complexity of the $DOAC$ policy scales as $O(MN2^N)$, where M is the number of points required to solve a one-dimensional search. Hence, I propose a suboptimal policy with a complexity of $O(N\log(N))$ that does not sacrifice the performance. Extensive simulation results showed the robustness of the $DOAC$ policy against CSI estimation errors.

Finally, I modeled the problem using hard-deadline framework where a strict deadline was imposed on each packet. As long as the percentage of packets that are transmitted by these deadlines exceed a prespecified threshold, the corresponding user is considered satisfied. I presented a potential joint power allocation and scheduling algorithm to this problem that works in the presence of both real-time and non-real-time users. The power allocation proposed for both kinds of users are in closed-form expressions. The scheduling algorithm is shown to have a linear complexity, in the number of users, for on-off fading channels and a strictly sub-exponential for continuous ones.

REFERENCES

[1] I. F. Akyildiz, W.-Y. Lee, M. C. Vuran, and S. Mohanty, "Next generation/dynamic spectrum access/cognitive radio wireless networks: a survey," *Comput. Netw.*, vol. 50, no. 13, pp. 2127–2159, Sep. 2006. [Online]. Available: http://dx.doi.org/10.1016/j.comnet.2006.05.001

[2] A. Ewaisha, A. Sultan, and T. ElBatt, "Optimization of channel sensing time and order for cognitive radios," in *Wireless Communications and Networking Conference (WCNC), 2011 IEEE*, March 2011, pp. 1414–1419.

[3] "Federal Communications Commission Spectrum Policy Task Force Report," 2002. [Online]. Available: http://transition.fcc.gov/sptf/files/SEWGFinalReport_1.pdf

[4] A. Asadi and V. Mancuso, "A survey on opportunistic scheduling in wireless communications," *Communications Surveys & Tutorials, IEEE*, vol. 15, no. 4, pp. 1671–1688, 2013.

[5] V. Asghari and S. Aissa, "Adaptive rate and power transmission in spectrum-sharing systems," *IEEE Transactions on Wireless Communications*, vol. 9, no. 10, pp. 3272 –3280, Oct 2010.

[6] K. Hamdi, W. Zhang, and K. Letaief, "Uplink scheduling with qos provisioning for cognitive radio systems," in *Wireless Communications and Networking Conference, 2007.WCNC 2007. IEEE*, march 2007, pp. 2592 –2596.

[7] Z. Guan, T. Melodia, and G. Scutari, "To transmit or not to transmit? distributed queueing games in infrastructureless wireless networks," *Networking, IEEE/ACM Transactions on*, vol. PP, no. 99, pp. 1–14, 2015.

[8] Y. Zhang and C. Leung, "Resource allocation in an OFDM-based cognitive radio system," *IEEE Transactions on Communications*, vol. 57, no. 7, pp. 1928–1931, July 2009.

[9] S. Wang, Z.-H. Zhou, M. Ge, and C. Wang, "Resource allocation for heterogeneous cognitive radio networks with imperfect spectrum sensing," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 3, pp. 464–475, March 2013.

[10] S. Shakkottai and R. Srikant, "Scheduling real-time traffic with deadlines over a wireless channel," *Wireless Networks*, vol. 8, no. 1, pp. 13–26, 2002.

[11] X. Kang, W. Wang, J. J. Jaramillo, and L. Ying, "On the performance of largest-deficit-first for scheduling real-time traffic in wireless networks," in *Proceedings of the fourteenth ACM international symposium on Mobile ad hoc networking and computing*. ACM, 2013, pp. 99–108.

[12] "IEEE standard for information technology–telecommunications and information exchange between systems local and metropolitan area networks–specific requirements part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications," *IEEE Std 802.11-2012 (Revision of IEEE Std 802.11-2007)*, pp. 1–2793, March 2012.

[13] I.-H. Hou, V. Borkar, and P. Kumar, "A theory of QoS for wireless," in *INFOCOM 2009, IEEE*, April 2009, pp. 486–494.

[14] Q. Zhao, L. Tong, A. Swami, and Y. Chen, "Decentralized cognitive MAC for opportunistic spectrum access in ad hoc networks: A POMDP framework," *Selected Areas in Communications, IEEE Journal on*, vol. 25, no. 3, pp. 589 –600, Apr 2007.

[15] H. Jiang, L. Lai, R. Fan, and H. Poor, "Optimal Selection of Channel Sensing Order in Cognitive Radio," *Wireless Communications, IEEE Transactions on*, vol. 8, no. 1, pp. 297 –307, Jan. 2009.

[16] A. Sabharwal, A. Khoshnevis, and E. Knightly, "Opportunistic Spectral Usage: Bounds and a Multi-Band CSMA/CA Protocol," *Networking, IEEE/ACM Transactions on*, vol. 15, no. 3, pp. 533–545, 2007.

[17] J. Jia, Q. Zhang, and X. Shen, "HC-MAC: A Hardware-Constrained Cognitive MAC for Efficient Spectrum Management," *Selected Areas in Communications, IEEE Journal on*, vol. 26, no. 1, pp. 106–117, 2008.

[18] H. T. Cheng and W. Zhuang, "Simple Channel Sensing Order in Cognitive Radio Networks," *Selected Areas in Communications, IEEE Journal on*, vol. 29, no. 4, pp. 676–688, 2011.

[19] E. PEI, J. LI, and F. CHENG, "Sensing-Throughput Tradeoff for Cognitive Radio Networks with Additional Primary Transmission Protection," *Journal of Computational Information Systems*, vol. 9, no. 10, pp. 3767–3773, 2013.

[20] S. Hu, Y.-D. Yao, and Z. Yang, "Cognitive medium access control protocols for secondary users sharing a common channel with time division multiple access primary users," *Wireless Communications and Mobile Computing*, vol. 14, no. 2, pp. 284–296, 2014. [Online]. Available: http://dx.doi.org/10.1002/wcm.2185

[21] C.-p. Li and M. J. Neely, "Delay and Power-Optimal Control in Multi-Class Queueing Systems," *ArXiv e-prints*, Jan. 2011.

[22] K. Jagannathan, M. Markakis, E. Modiano, and J. Tsitsiklis, "Throughput optimal scheduling over time-varying channels in the presence of heavy-tailed traffic," *Information Theory, IEEE Transactions on*, vol. 60, no. 5, pp. 2896–2909, May 2014.

[23] M. J. Neely, E. Modiano, and C. E. Rohrs, "Power allocation and routing in multibeam satellites with time-varying channels," *IEEE/ACM Transactions on Networking*, vol. 11, no. 1, pp. 138–152, 2003.

[24] J. Wang, A. Huang, L. Cai, and W. Wang, "On the queue dynamics of multiuser multichannel cognitive radio networks," *Vehicular Technology, IEEE Transactions on*, vol. 62, no. 3, pp. 1314–1328, March 2013.

[25] M. M. Rashid, M. J. Hossain, E. Hossain, and V. K. Bhargava, "Opportunistic spectrum scheduling for multiuser cognitive radio: a queueing analysis," *Wireless Communications, IEEE Transactions on*, vol. 8, no. 10, pp. 5259–5269, 2009.

[26] Z. Li, C. Yin, and G. Yue, "Delay-bounded power-efficient packet scheduling for uplink systems of lte," in *Wireless Communications, Networking and Mobile Computing, 2009. WiCom '09. 5th International Conference on*, Sept 2009, pp. 1–4.

[27] A. E. Ewaisha and C. Tepedelenliolu, "Throughput optimization in multichannel cognitive radios with hard-deadline constraints," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 4, pp. 2355–2368, April 2016.

[28] W. K. Lai and C.-L. Tang, "Qos-aware downlink packet scheduling for {LTE} networks," *Computer Networks*, vol. 57, no. 7, pp. 1689 – 1698, 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128613000352

[29] G. Piro, L. A. Grieco, G. Boggia, R. Fortuna, and P. Camarda, "Two-level downlink scheduling for real-time multimedia services in lte networks," *IEEE Transactions on Multimedia*, vol. 13, no. 5, pp. 1052–1065, 2011.

[30] K. S. Kim, C. p. Li, and E. Modiano, "Scheduling multicast traffic with deadlines in wireless networks," in *IEEE INFOCOM 2014 - IEEE Conference on Computer Communications*, April 2014, pp. 2193–2201.

[31] P.-C. Hsieh and I.-H. Hou, "Heavy-traffic analysis of QoE optimality for on-demand video streams over fading channels."

[32] A. Ewaisha and C. Tepedelenlioğlu, "Joint scheduling and power-control for delay guarantees in heterogeneous cognitive radios," *arXiv preprint arXiv:1602.08010*, 2015.

[33] I.-H. Hou, P.-C. Hsieh *et al.*, "Qoe-optimal scheduling for on-demand video streams over unreliable wireless networks," in *Proceedings of the 16th ACM International Symposium on Mobile Ad Hoc Networking and Computing*. ACM, 2015, pp. 207–216.

[34] I.-H. Hou and P. Kumar, "Scheduling heterogeneous real-time traffic over fading wireless channels," in *INFOCOM, 2010 Proceedings IEEE*, March 2010, pp. 1–9.

[35] I.-H. Hou, "Broadcasting delay-constrained traffic over unreliable wireless links with network coding," *IEEE/ACM Trans. Netw.*, vol. 23, no. 3, pp. 728–740, Jun. 2015. [Online]. Available: http://dx.doi.org/10.1109/TNET.2014.2304880

[36] L. Yang, Y. E. Sagduyu, and J. H. Li, "Adaptive network coding for scheduling real-time traffic with hard deadlines," in *Proceedings of the thirteenth ACM international symposium on Mobile Ad Hoc Networking and Computing*, ser. MobiHoc '12. New York, NY, USA: ACM, 2012, pp. 105–114. [Online]. Available: http://doi.acm.org/10.1145/2248371.2248389

[37] J. J. Jaramillo and R. Srikant, "Optimal scheduling for fair resource allocation in ad hoc networks with elastic and inelastic traffic," in *INFOCOM, 2010 Proceedings IEEE*. IEEE, 2010, pp. 1–9.

[38] A. Ewaisha and C. Tepedelenlioğlu, "Throughput Maximization in Multichannel Cognitive radio Systems with Delay Constraints," in *the 47th Asilomar Conference on Signals, Systems, and Computers, 2013. IEEE*, November 2013.

[39] G. Zhang, X. Wang, Y.-C. Liang, and J. Liu, "Fast and robust spectrum sensing via kolmogorov-smirnov test," *Communications, IEEE Transactions on*, vol. 58, no. 12, pp. 3410–3416, December 2010.

[40] Y. Zou, Y.-D. Yao, and B. Zheng, "Cognitive transmissions with multiple relays in cognitive radio networks," *Wireless Communications, IEEE Transactions on*, vol. 10, no. 2, pp. 648–659, 2011.

[41] G. Ding, J. Wang, Q. Wu, L. Zhang, Y. Zou, Y.-D. Yao, and Y. Chen, "Robust spectrum sensing with crowd sensors," *Communications, IEEE Transactions on*, vol. 62, no. 9, pp. 3129–3143, Sept 2014.

[42] Y.-C. Liang, Y. Zeng, E. C. Peh, and A. T. Hoang, "Sensing-throughput tradeoff for cognitive radio networks," *Wireless Communications, IEEE Transactions on*, vol. 7, no. 4, pp. 1326–1337, 2008.

[43] Y. Zou, Y.-D. Yao, and B. Zheng, "Outage probability analysis of cognitive transmissions: impact of spectrum sensing overhead," *Wireless Communications, IEEE Transactions on*, vol. 9, no. 8, pp. 2676–2688, 2010.

[44] A. Goldsmith, *Wireless Communications.* New York, NY, USA: Cambridge University Press, 2005.

[45] S. Boyd and L. Vandenberghe, *Convex Optimization.* New York, NY, USA: Cambridge University Press, 2004.

[46] E. Miersemann, *Calculus of Variations*, October, 2012.

[47] R. M. Corless, G. H. Gonnet, D. E. G. Hare, D. J. Jeffrey, and D. E. Knuth, "On the Lambert W Function," in *ADVANCES IN COMPUTATIONAL MATHEMATICS*, 1996, pp. 329–359.

[48] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes 3rd Edition: The Art of Scientific Computing*, 3rd ed. New York, NY, USA: Cambridge University Press, 2007.

[49] D. G. Luenberger and Y. Ye, *Linear and Nonlinear Programming*, 3rd ed., ser. International Series in Operations Research & Management Science, 116. Springer, New York, 2008.

[50] D. Furth, D. Conley, C. Murphy, B. Romano, and S. Stone, "Federal communications commission spectrum policy task force report of the spectrum rights and responsibilities working group," *Spectrum*, 2002. [Online]. Available: http://www.fcc.gov/sptf/reports.html

[51] E. Axell, G. Leus, E. Larsson, and H. Poor, "Spectrum sensing for cognitive radio : State-of-the-art and recent advances," *Signal Processing Magazine, IEEE*, vol. 29, no. 3, pp. 101–116, 2012.

[52] T. Yucek and H. Arslan, "A survey of spectrum sensing algorithms for cognitive radio applications," *Communications Surveys Tutorials, IEEE*, vol. 11, no. 1, pp. 116 –130, quarter 2009.

[53] D. Tse and P. Viswanath, "Fundamentals of wireless communications," 2004.

[54] Y. Zhang *et al.*, "Asymptotic ber analysis of a simo multiuser diversity system," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 9, pp. 5330–5335, 2009.

[55] S. Haykin, "Cognitive radio: brain-empowered wireless communications," *Selected Areas in Communications, IEEE Journal on*, vol. 23, no. 2, pp. 201–220, 2005.

[56] D. Bertsekas and R. Gallager, *Data Networks (2Nd Ed.).* Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1992.

[57] F. Semiconductor, "Long term evolution protocol overview," *White Paper, Document No. LTEPTCLOVWWP, Rev 0 Oct*, 2008.

[58] M. J. Neely, "Dynamic power allocation and routing for satellite and wireless networks with time varying channels," Ph.D. dissertation, Citeseer, 2003.

[59] D. D. Yao, "Dynamic scheduling via polymatroid optimization," in *Performance Evaluation of Complex Systems: Techniques and Tools.* Springer, 2002, pp. 89–113.

[60] R. Srikant and L. Ying, *Communication Networks: An Optimization, Control, and Stochastic Networks Perspective.* Cambridge University Press, 2013.

[61] L. Georgiadis, M. J. Neely, and L. Tassiulas, *Resource allocation and cross-layer control in wireless networks.* Now Publishers Inc, 2006.

[62] R. Urgaonkar, B. Urgaonkar, M. J. Neely, and A. Sivasubramaniam, "Optimal power cost management using stored energy in data centers," in *Proceedings of the ACM SIGMETRICS Joint International Conference on Measurement and Modeling of Computer Systems.* New York, NY, USA: ACM, 2011, pp. 221–232. [Online]. Available: http://doi.acm.org/10.1145/1993744.1993766

[63] A. Rajan, "On the ordering of communication channels," Ph.D. dissertation, Arizona State University, 2014.

[64] M. DeGroot and M. Schervish, *Probability and Statistics*, 4th ed. Pearson Education, 2011. [Online]. Available: https://books.google.com/books?id=XHmitwAACAAJ

APPENDIX A

PROOF OF LEMMA 1

Proof. We carry out the proof by contradiction. Assume, for some i, that $\gamma^*_{\mathrm{th}}(i) < \lambda^*_{\mathrm{P}}$. Thus the reward starting from channel i, $U_i\left([\gamma^*_{\mathrm{th}}(i), \gamma^*_{\mathrm{th}}(i+1), ..., \gamma^*_{\mathrm{th}}(M)]^T, \mathbf{P}^*_i\right)$, becomes

$$\begin{aligned}
&\theta_i c_i \int_{\gamma^*_{\mathrm{th}}(i)}^{\infty} \log(1+P^*_{1,i}\gamma) f_\gamma(\gamma)\, d\gamma + \theta_i U^*_{i+1} \int_0^{\gamma^*_{\mathrm{th}}(i)} f_\gamma(\gamma)\, d\gamma \\
&\qquad + (1-\theta_i) U^*_{i+1} &\text{(A.1)} \\
\leq &\theta_i c_i \int_{\lambda^*_{\mathrm{P}}}^{\infty} \log(1+P^*_{1,i}\gamma) f_\gamma(\gamma)\, d\gamma + \theta_i U^*_{i+1} \int_0^{\lambda^*_{\mathrm{P}}} f_\gamma(\gamma)\, d\gamma \\
&\qquad + (1-\theta_i) U^*_{i+1} &\text{(A.2)} \\
= &U_i\left([\lambda^*_{\mathrm{P}}, \gamma^*_{\mathrm{th}}(i+1), ..., \gamma^*_{\mathrm{th}}(M)]^T, \mathbf{P}^*_i\right). &\text{(A.3)}
\end{aligned}$$

Where inequality (A.2) follows by adding the term $\theta_i \left(\int_{\gamma^*_{\mathrm{th}}(i)}^{\lambda^*_{\mathrm{P}}} f_\gamma(\gamma)\, d\gamma\right) U^*_{i+1}$ to (A.1) while (A.3) follows by the definition of the right-hand-side of (A.2). Using equation (2.2), we can calculate the reward U_{i-1} for both the left-hand-side and right-hand-side of the previous inequality. Thus the following inequality holds

$$\begin{aligned}
&U_{i-1}\left([\gamma^*_{\mathrm{th}}(i-1), \gamma^*_{\mathrm{th}}(i), ..., \gamma^*_{\mathrm{th}}(M)]^T, \mathbf{P}^*_{i-1}\right) \leq \\
&U_{i-1}\left([\gamma^*_{\mathrm{th}}(i-1), \lambda^*_{\mathrm{P}}, ..., \gamma^*_{\mathrm{th}}(M)]^T, \mathbf{P}^*_{i-1}\right). \qquad \text{(A.4)}
\end{aligned}$$

Carrying out the last step recursively $i-2$ more times, we find the relation

$$\begin{aligned}
&U_1\left([\gamma^*_{\mathrm{th}}(1), ..., \gamma^*_{\mathrm{th}}(i-1), \gamma^*_{\mathrm{th}}(i), ..., \gamma^*_{\mathrm{th}}(M)]^T, \mathbf{P}^*_1\right) \leq \\
&U_1\left([\gamma^*_{\mathrm{th}}(1), ..., \gamma^*_{\mathrm{th}}(i-1), \lambda^*_{\mathrm{P}}, ..., \gamma^*_{\mathrm{th}}(M)]^T, \mathbf{P}^*_1\right), \qquad \text{(A.5)}
\end{aligned}$$

which contradicts with the fact that $\gamma^*_{\mathrm{th}}(i)$ is optimal. □

APPENDIX B

PROOF OF THEOREM 1

Proof. We first get S_i^*, U_i^* and p_i^* by substituting by equations $\gamma_{\text{th}}^*(i)$ and $P_{1,i}^*(\gamma)$ in the three equations (2.1), (2.2) and (2.3), respectively. Then we differentiate with respect to λ_{P}^*, treating λ_{D}^* as a constant, yielding

$$\begin{aligned}\frac{\partial S_i^*}{\partial \lambda_{\text{P}}^*} =& -\theta_i f_\gamma(\gamma_{\text{th}}^*(i))\frac{\partial \gamma_{\text{th}}^*(i)}{\partial \lambda_{\text{P}}^*}\left(c_i P_i^*(\gamma_{\text{th}}^*(i)) - S_{i+1}^*\right) - \\ & \theta_i c_i \frac{\bar{F}_\gamma(\gamma_{\text{th}}^*(i))}{(\lambda_{\text{P}}^*)^2} + \left(1-\theta_i \bar{F}_\gamma(\gamma_{\text{th}}^*(i))\right)\frac{\partial S_{i+1}^*}{\partial \lambda_{\text{P}}^*}, \qquad \text{(B.1)}\\ \frac{\partial U_i^*}{\partial \lambda_{\text{P}}^*} =& -\theta_i f_\gamma(\gamma_{\text{th}}^*(i))\frac{\partial \gamma_{\text{th}}^*(i)}{\partial \lambda_{\text{P}}^*}\times \\ & \left[\lambda_{\text{P}}^*\left(c_i P_i^*(\gamma_{\text{th}}^*(i)) - S_{i+1}^*\right) - \lambda_{\text{D}}^*\left(1-p_{i+1}^*\right)\right] - \\ & \theta_i c_i \frac{\bar{F}_\gamma(\gamma_{\text{th}}^*(i))}{\lambda_{\text{P}}^*} + \left(1-\theta_i \bar{F}_\gamma(\gamma_{\text{th}}^*(i))\right)\frac{\partial U_{i+1}^*}{\partial \lambda_{\text{P}}^*}, \qquad \text{(B.2)}\\ \frac{\partial p_i^*}{\partial \lambda_{\text{P}}^*} =& -\theta_i f_\gamma(\gamma_{\text{th}}^*(i))\frac{\partial \gamma_{\text{th}}^*(i)}{\partial \lambda_{\text{P}}^*}\left(1-p_{i+1}^*\right) + \qquad \text{(B.3)}\\ & \left(1-\theta_i \bar{F}_\gamma(\gamma_{\text{th}}^*(i))\right)\frac{\partial p_{i+1}^*}{\partial \lambda_{\text{P}}^*}, \qquad \text{(B.4)}\end{aligned}$$

respectively. Multiplying equation (B.1) by $-\lambda_{\text{P}}^*$ and equation (B.3) by λ_{D}^* then adding them to equation (B.2) we can easily show that, for all $i \in \mathcal{M}$,

$$\frac{\partial U_i^*}{\partial \lambda_{\text{P}}^*} - \lambda_{\text{P}}\frac{\partial S_i^*}{\partial \lambda_{\text{P}}^*} + \lambda_{\text{D}}\frac{\partial p_i^*}{\partial \lambda_{\text{P}}^*} = 0. \qquad \text{(B.5)}$$

We now find the derivative of $\gamma_{\text{th}}^*(i)$ with respect to λ_{P}^* by differentiating both sides of equation (2.7) with respect to λ_{P}^*, while treating λ_{D}^* as a constant,then using equation (B.5), then rearranging we get

$$\frac{\partial \gamma_{\text{th}}^*(i)}{\partial \lambda_{\text{P}}^*} = \frac{c_i P_i^*(\gamma_{\text{th}}^*(i)) - S_{i+1}^*}{c_i \frac{\lambda_{\text{P}}^*}{\gamma_{\text{th}}^*(i)} P_i^*(\gamma_{\text{th}}^*(i))}. \qquad \text{(B.6)}$$

Substituting by equation (B.6) in (B.1) we get

$$\begin{aligned}\frac{\partial S_i^*}{\partial \lambda_{\text{P}}^*} =& -\alpha_i\left[c_i P_i^*(\gamma_{\text{th}}^*(i)) - S_{i+1}^*\right]^2 - \theta_i c_i \frac{\bar{F}_\gamma(\gamma_{\text{th}}^*(i))}{(\lambda_{\text{P}}^*)^2} + \\ & \left(1-\theta_i \bar{F}_\gamma(\gamma_{\text{th}}^*(i))\right)\frac{\partial S_{i+1}^*}{\partial \lambda_{\text{P}}^*}, \qquad \text{(B.7)}\end{aligned}$$

where α_i is given by

$$\alpha_i = \frac{\theta_i f_\gamma(\gamma^*_{\text{th}}(i))}{c_i \frac{\lambda^*_{\text{P}}}{\gamma^*_{\text{th}}(i)} P^*_i(\gamma^*_{\text{th}}(i))} \geq 0, \tag{B.8}$$

Now evaluating (B.7) at $i = M$ and $i = M - 1$ we get

$$\frac{\partial S^*_M}{\partial \lambda^*_{\text{P}}} = -\alpha_M \left[c_M P^*_M(\gamma^*_{\text{th}}(M))\right]^2 - \theta_M c_M \frac{\bar{F}_\gamma(\gamma^*_{\text{th}}(M))}{(\lambda^*_{\text{P}})^2}, \tag{B.9}$$

$$\begin{aligned}\text{and } \frac{\partial S^*_{M-1}}{\partial \lambda^*_{\text{P}}} &= -\alpha_{M-1}\left[c_{M-1} P^*_{M-1}(\gamma^*_{\text{th}}(M-1)) - S^*_M\right]^2 \\ &\quad - \theta_{M-1} c_{M-1} \frac{\bar{F}_\gamma(\gamma^*_{\text{th}}(M-1))}{(\lambda^*_{\text{P}})^2} \\ &\quad + \left(1 - \theta_{M-1}\bar{F}_\gamma(\gamma^*_{\text{th}}(M-1))\right)\frac{\partial S^*_M}{\partial \lambda^*_{\text{P}}},\end{aligned} \tag{B.10}$$

respectively. We can see that $\frac{\partial S^*_M}{\partial \lambda^*_{\text{P}}} < 0$, hence $\frac{\partial S^*_{M-1}}{\partial \lambda^*_{\text{P}}} < 0$. By induction, let's assume that $\frac{\partial S^*_{i+1}}{\partial \lambda^*_{\text{P}}} < 0$. From (B.7) we get that

$$\begin{aligned}\frac{\partial S^*_i}{\partial \lambda^*_{\text{P}}} =& -\alpha_i\left(c_i P^*_i(\gamma^*_{\text{th}}(i)) - S^*_{i+1}\right)^2 - \theta_i c_i \frac{\bar{F}_\gamma(\gamma^*_{\text{th}}(i))}{(\lambda^*_{\text{P}})^2} + \\ &\left(1 - \theta_i \bar{F}_\gamma(\gamma^*_{\text{th}}(i))\right)\frac{\partial S^*_{i+1}}{\partial \lambda^*_{\text{P}}} < 0\end{aligned} \tag{B.11}$$

since all its terms are negative. Finally we find that $\frac{\partial S^*_1}{\partial \lambda^*_{\text{P}}} < 0$ indicating that S^*_1 is monotonically decreasing in λ^*_{P} given any fixed $\lambda^*_{\text{D}} \geq 0$.

Now, to get an upper bound on λ^*_{P}, we know that

$$S^*_i = \theta_i c_i \int_{\gamma^*_{\text{th}}(i)}^{\infty} \left(\frac{1}{\lambda^*_{\text{P}}} - \frac{1}{\gamma}\right) f_\gamma(\gamma)\, d\gamma + \left[1 - \theta_i \bar{F}_\gamma(\gamma^*_{\text{th}}(i))\right] S^*_{i+1}. \tag{B.12}$$

We can upper bound the first term in (B.12) by $\theta_i c_i/\lambda^*_{\text{P}}$, while $\left[1 - \theta_i \bar{F}_\gamma(\gamma^*_{\text{th}}(i))\right] < 1$. Using these two bounds we can write $S^*_1 < \sum_{i=1}^{M} \theta_i c_i/\lambda^*_{\text{P}}$. But since $S^*_1 = P_{\text{avg}}$, the upper bound on λ^*_{P}, mentioned in Theorem 1, follows. $\square$

APPENDIX C

PROOF OF LEMMA 2

Proof. We provide a proof sketch for this bound. We know that at the optimal point $p_1^* = \frac{1}{\bar{D}_{\max}}$ and that $p_1^* = \theta_1 \bar{F}_\gamma\left(\gamma_{\mathrm{th}}^*\left(1\right)\right) + \left(1 - \theta_1 \bar{F}_\gamma\left(\gamma_{\mathrm{th}}^*\left(1\right)\right)\right) p_2^*$. But since the second term in the latter equation is always positive, then

$$\theta_1 \bar{F}_\gamma\left(\gamma_{\mathrm{th}}^*\left(1\right)\right) < \frac{1}{\bar{D}_{\max}}. \tag{C.1}$$

Substituting by (2.12) in (C.1) and rearranging we can upper bound λ_{D}^* by

$$\frac{c_1\left(\log\left(\frac{\lambda_{\mathrm{P}}^*}{\bar{F}_\gamma^{-1}\left(\frac{1}{\theta_1 \bar{D}_{\max}}\right)}\right) - \frac{\lambda_{\mathrm{P}}^*}{\bar{F}_\gamma^{-1}\left(\frac{1}{\theta_1 \bar{D}_{\max}}\right)} + 1\right) + U_2^* - \lambda_{\mathrm{P}}^* S_2^*}{1 - p_2^*}$$

We can easily upper bound $\log\left(\lambda_{\mathrm{P}}^* / \bar{F}_\gamma^{-1}\left(1/\left(\theta_1 \bar{D}_{\max}\right)\right)\right) - \lambda_{\mathrm{P}}^* / \bar{F}_\gamma^{-1}\left(1/\left(\theta_1 \bar{D}_{\max}\right)\right)$ by substituting $\lambda_{\mathrm{P}}^{\max}$ for λ_{P}^* when $\lambda_{\mathrm{P}}^* < \bar{F}_\gamma^{-1}\left(1/\left(\theta_1 \bar{D}_{\max}\right)\right)$ and by 1 otherwise. Moreover, it can also be shown that $U_2^* < U_2^{\max}$, $p_2^* < p_2^{\max}$ and that $\lambda_{\mathrm{P}}^* S_2^* > 0$ and from Theorem 1 we have $\lambda_{\mathrm{P}}^* < \lambda_{\mathrm{P}}^{\max}$, the proof then follows. $\square$

APPENDIX D

PROOF OF THEOREM 3

Proof. In this proof, we show that the drift-plus-penalty under this algorithm is upper bounded by some constant, which indicates that the virtual queues are mean rate stable [61, 62].

We define the Lyapunov function as $L(k) \triangleq \frac{1}{2}\sum_{i=1}^{N} Y_i^2(k)$ and Lyapunov drift to be

$$\Delta(k) \triangleq \mathbb{E}_{\mathbf{Y}(k)}\left[L(k+1) - L(k)\right], \tag{D.1}$$

Squaring equation (3.10) then taking the conditional expectation we can write the following bound

$$\frac{1}{2}\,\mathbb{E}_{\mathbf{Y}(k)}\left[Y_i^2(k+1) - Y_i^2(k)\right] \leq Y_i(k)\,\mathbb{E}_{\mathbf{Y}(k)}\left[T_k\right]\lambda_i\left(\mathbb{E}_{\mathbf{Y}(k)}\left[W_i^{(j)}\right] - r_i(k)\right) + C_{Y_i}. \tag{D.2}$$

where we use the bound $\mathbb{E}_{\mathbf{Y}(k)}\left[\left(\sum_{j\in\mathcal{A}_i(k)} W_i^{(j)}\right)^2\right] + \mathbb{E}_{\mathbf{Y}(k)}\left[\left(\sum_{j\in\mathcal{A}_(k)} r_i(k)\right)^2\right] < C_{Y_i}$. The derivation is similar to that in [21, Lemma7]. Given some fixed control parameter $V > 0$, we add the penalty term $V\sum_i \mathbb{E}_{\mathbf{Y}(k)}\left[r_i(k)T_k\right]$ to both sides of (D.1). Using the bound in (D.2) the drift-plus-penalty term becomes bounded by

$$\Delta\left(\mathbf{U}(k)\right) + V\sum_{i=1}^{N}\mathbb{E}_{\mathbf{Y}(k)}\left[r_i(k)T_k\right] \leq C_Y + \mathbb{E}_{\mathbf{Y}(k)}\left[T_k\right]\Phi \quad \text{where} \tag{D.3}$$

$$\Phi \triangleq \sum_{i=1}^{N}\left(V - Y_i(k)\lambda_i\right) r_i(k) + \sum_{j=1}^{N} Y_{\pi_j}(k)\lambda_{\pi_j}\,\mathbb{E}_{\mathbf{Y}(k)}\left[W_{\pi_j}^{(j)}\right], \tag{D.4}$$

We define the *DOIC* policy to be the policy that finds the values of $\boldsymbol{\pi}(k)$, $\{\mathbf{P}^{(t)}\}$ and $\mathbf{r}(k)$ vector that minimize Φ subject to the instantaneous interference, the maximum power and the single-SU-per-time-slot constraints in problem (3.6). We can observe that the variables $\mathbf{r}(k)$, $\{\mathbf{P}^{(t)}\}$ and $\boldsymbol{\pi}(k)$ can be chosen independently from each other. Step 4.a in the *DOIC* policy finds the optimum value of $r_i(k)$, $\forall i \in \mathcal{N}$. Moreover, since $\mathbb{E}_{\mathbf{Y}(k)}\left[W_i^{(j)}\right]$ is decreasing in $P_i^{(t)}$ $\forall t \in \mathcal{F}(k)$, the optimum value for $P_i^{(t)}$ is equation (3.15). Finally, from [59] the $c\mu$-rule can be applied to find the optimum priority list $\boldsymbol{\pi}(k)$ which is given by Step 1 in the *DOIC* policy.

Now, since the proposed *DOIC* policy minimizes Φ, this gives a lower bound on Φ compared to any other policy including the optimal policy that solves (3.6). Hence, we now evaluate Φ at the optimal policy that solves (3.6) with the help of a genie-aided knowledge of $r_i(k) = \bar{W}_i^*$ yielding $\Phi^{\text{opt}} = V\sum_{i=1}^{N}\bar{W}_i^*$, where we use $\mathbb{E}_{\mathbf{Y}(k)}\left[W_i^{(j)}\right] = \bar{W}_i^*$. Substituting by Φ^{opt} in the right-hand-side (r.h.s.) of (D.3) gives an upper bound on the drift-plus-penalty when evaluated at the *DOIC* policy. Namely

$$\Delta\left(\mathbf{Y}(k)\right) + V\sum_{i=1}^{N}\mathbb{E}_{\mathbf{Y}(k)}\left[r_i(k)T_k\right] \leq C_Y + V\sum_{i=1}^{N}\bar{W}_i^*\,\mathbb{E}_{\mathbf{Y}(k)}\left[T_k\right] \tag{D.5}$$

Taking $\mathbb{E}\left[\cdot\right]$, summing over $k = 0, \cdots, K-1$, denoting $\mathbf{Y}_i(0) \triangleq 0$ for all $i \in \mathcal{N}$, and dividing by $V\sum_{k=0}^{K-1}\mathbb{E}\left[T_k\right]$ we get

$$\sum_{i=1}^{N}\frac{\mathbb{E}\left[Y_i^2(K)\right]}{\sum_{k=0}^{K-1}\mathbb{E}\left[T_k\right]} + \sum_{i=1}^{N}\frac{\sum_{k=0}^{K-1}\mathbb{E}\left[r_i(k)T_k\right]}{\sum_{k=0}^{K-1}\mathbb{E}\left[T_k\right]} \overset{(a)}{\leq} \frac{aC_Y}{V} + \sum_{i=1}^{N}\bar{W}_i^* \triangleq C_1. \tag{D.6}$$

where in the r.h.s. of inequality (a) we used $\mathbb{E}\left[T_k\right] \geq \mathbb{E}\left[I(k)\right] = 1/a$, and C_1 is some constant that is not a function in K. To prove the mean rate stability of the sequence $\{Y_i(k)\}_{k=0}^{\infty}$ for any $i \in \mathcal{N}$, we remove the first and third terms in the left-side of (D.6) as well as the summation operator from the second term to obtain $\mathbb{E}\left[Y_i^2(K)\right]/K \leq C_1$ $\forall i \in \mathcal{N}$. Using Jensen's inequality we note that

$$\frac{\mathbb{E}\left[Y_i(K)\right]}{K} \leq \sqrt{\frac{\mathbb{E}\left[Y_i^2(K)\right]}{K^2}} \leq \sqrt{\frac{C_1}{K}}. \tag{D.7}$$

Finally, taking the limit when $K \to \infty$ completes the mean rate stability proof. On the other hand, to prove the upper bound in Theorem 3, we use the fact that $r_i(k)$ and $|\mathcal{A}_i(k)|$ are independent random variables (see step 4-a in *DOIC*) to replace $\mathbb{E}\left[|\mathcal{A}_i(k)|r_i(k)\right]$ by $\lambda_i\,\mathbb{E}\left[T_k r_i(k)\right]$ in equation (3.13), then we take the limit of (3.13) as $K \to \infty$, use the mean rate stability theorem and sum over $i \in \mathcal{N}$ to get

$$\sum_{i=1}^{N}\frac{\mathbb{E}\left[\sum_{k=0}^{K-1}\left(\sum_{j\in\mathcal{A}_i(k)}W_i^{(j)}\right)\right]}{\mathbb{E}\left[\sum_{k=0}^{K-1}|\mathcal{A}_i(k)|\right]} \leq \sum_{i=1}^{N}\frac{\sum_{k=0}^{K-1}\mathbb{E}\left[r_i(k)T_k\right]}{\sum_{k=0}^{K-1}\mathbb{E}\left[T_k\right]} \overset{(b)}{\leq} \frac{aC_Y}{V} + \sum_{i=1}^{N}\bar{W}_i^*, \tag{D.8}$$

where inequality (b) comes from removing the first summation in the left-side of (D.6). Taking the limit when $K \to \infty$ and using equation (3.9) completes the proof. $\square$

APPENDIX E

EXISTENCE OF SERVICE TIME MOMENTS

Lemma 10. *Given any distribution for $P_i^{(t)}\gamma_i^{(t)}$ the inequality $\mathbb{E}\left[s_i^n\right] < \infty$ holds $\forall n \geq 1$. Moreover, when the power is given by $P_i^{(t)} = \min\left(I_{\text{inst}}/g_i^{(t)}, P\right)$ for some fixed parameter $P \in [P_i^{\min}, P_{\max}]$, the inequality $\mathbb{E}\left[s_i^2\right] \leq \left(L^2 + L\left(1 - p_i(P_i^{\min})\right)\right)/p_i^2(P_i^{\min})$ holds with $p_i(P) \triangleq 1 - \mathbf{Pr}\left[R_i(P) = 0\right]$.*

Proof. We carry out the proof by bounding the moments of s_i by the respective moments of the random variable $s_{\mathrm{B},i}$ which is the service time for a system with a binary transmission rate, i.e. a system with $R_i^{(t)} \in \{0, 1\}$. The proof of the first part of the lemma follows by showing that all the moments of $s_{\mathrm{B},i}$ are finite. The second part of the lemma is a special case where we set $P_i^{(t)} = \min\left(I_{\text{inst}}/g_i^{(t)}, P\right)$.

In this proof we drop the index i for simplicity whenever it is clear from the context. Given some, possibly random, power allocation policy $P_i^{(t)}$ define the i.i.d. random process $R_{\mathrm{B},i}^{(t)} \in \{0, 1\}$, $t \geq 1$, with $\mathbf{Pr}\left[R_{\mathrm{B},i}^{(t)} = 0\right] = \mathbf{Pr}\left[R_i^{(t)} = 0\right]$. Dropping the index i, the following inequality holds for any $x \geq 1$

$$\mathbf{Pr}\left[\sum_{t=1}^{x} R_{\mathrm{B}}^{(t)} \leq L\right] \geq \mathbf{Pr}\left[\sum_{t=1}^{x} R^{(t)} \geq L\right], \tag{E.1}$$

which says that the probability of transmitting L bits or more in x time slots is higher if the transmission process is $R^{(t)}$ compared to the binary transmission process $R_{\mathrm{B}}^{(t)}$. Defining $s_{\mathrm{B}} \triangleq \{\min x : \sum_{t=1}^{x} R_{\mathrm{B}}^{(t)} \geq L\}$ as the binary service time which is the number of time slots required to transmit L bits given that the transmission process is $R_{\mathrm{B}}^{(t)}$, we can write

$$\mathbf{Pr}\left[s_{\mathrm{B}} \leq x\right] = \mathbf{Pr}\left[\sum_{t=1}^{x} R_{\mathrm{B}}^{(t)} \geq L\right] \tag{E.2}$$

$$\geq \mathbf{Pr}\left[\sum_{t=1}^{x} R^{(t)} \geq L\right] \tag{E.3}$$

$$= \mathbf{Pr}\left[s \leq x\right] \tag{E.4}$$

According to the theory of stochastic ordering, when two random variables have ordered cumulative distribution functions, their respective moments are ordered [63, equation (2.14) pp. 16]. In other words, if $\mathbf{Pr}[s \leq x] \geq \mathbf{Pr}[s_{\mathrm{B}} \leq x]$, then $\mathbb{E}[s^n] \leq \mathbb{E}[s_{\mathrm{B}}^n]$, $\forall n \geq 1$. It suffices to show that the moments of s_{B} are finite.

Define s_{NB} as a random variable following the negative binomial distribution [64, pp. 297] with success probability $1 - \mathbf{Pr}\left[R_{\mathrm{B}}^{(t)} = 0\right]$ while the number of successes equals L. s_{NB} refers to the number of time slots having $R_{\mathrm{B}}^{(t)} = 0$ before transmitting the Lth bit. We can see that $s_{\mathrm{B}} = s_{\mathrm{NB}} + L$. Thus we have

$$\mathbb{E}[s_{\mathrm{B}}^n] = \sum_{j=0}^{n} \binom{n}{j} \mathbb{E}\left[s_{\mathrm{NB}}^j\right] L^{n-j} < \infty, \tag{E.5}$$

where the inequality follows since all the moments of the negative binomial distribution exist [64, pp.297]. The first part of the lemma holds.

For the second part of the lemma, we set $P_i^{(t)} = \min\left(I_{\mathrm{inst}}/g_i^{(t)}, P\right)$ for some deterministic parameter $P \geq P_i^{\min}$ and define $p_i(P) \triangleq 1 - \mathbf{Pr}[R_i(P) = 0]$ with $R_i(P)$ defined in (3.14). Given the moment generating function of s_{NB} as [64, pp. 894]

$$\mathbb{E}\left[e^{x s_{\mathrm{NB}}}\right] = \frac{p_i^L(P)}{\left(1 - (1 - p_i(P) e^x)\right)^L}, \tag{E.6}$$

the first two moments of s_{NB} can be derived as

$$\mathbb{E}[s_{\mathrm{NB}}] = \frac{(1 - p_i(P))\, L}{p_i(P)}, \qquad \text{and} \tag{E.7}$$

$$\mathbb{E}\left[s_{\mathrm{NB}}^2\right] = \frac{(1 - p_i(P))^2 L^2 + (1 - p_i(P))\, L}{p_i^2(P)}. \tag{E.8}$$

These two moments can be shown to be decreasing in $p_i(P)$. The proof of the second part of the lemma follows using the bound $p_i(P) \geq p_i(P_i^{\min})$ and the inequality $\mathbb{E}[s^2] \leq \mathbb{E}[s_{\mathrm{B}}^2] = \mathbb{E}[s_{\mathrm{NB}}^2] + 2L\,\mathbb{E}[s_{\mathrm{NB}}] + L^2$. □

APPENDIX F

PROOF OF THEOREM 4

Proof. This proof is similar to that in Appendix D. We define $\mathbf{U}(k) \triangleq [X(k), \mathbf{Y}(k)]^T$, the Lyapunov function as $L(k) \triangleq \frac{1}{2}X^2(k) + \frac{1}{2}\sum_{i=1}^{N} Y_i^2(k)$ and Lyapunov drift to be

$$\Delta(k) \triangleq \mathbb{E}_{\mathbf{U}(k)}\left[L(k+1) - L(k)\right]. \tag{F.1}$$

Squaring equation (3.19) then taking the conditional expectation we can get the bound

$$\frac{\mathbb{E}_{\mathbf{U}(k)}\left[X^2(k+1) - X^2(k)\right]}{2} \leq C_X + X(k)\left(\mathbb{E}_{\mathbf{U}(k)}\left[\sum_{t\in\mathcal{F}(k)} P_i^{(t)} g_i^{(t)}\right] - I_{\text{avg}}\,\mathbb{E}_{\mathbf{U}(k)}\left[T_k\right]\right), \tag{F.2}$$

where we use the bound $\mathbb{E}_{\mathbf{U}(k)}\left[\left(\sum_{i=1}^{N}\sum_{t\in\mathcal{F}(k)} P_i^{(t)} g_i^{(t)}\right)^2 + \left(I_{\text{avg}}T_k\right)^2\right] < C_X$ in equation (F.2) and omit the derivation of this bound. Given some fixed control parameter $V > 0$, we add the penalty term $V\sum_i \mathbb{E}_{\mathbf{U}(k)}\left[r_i(k)T_k\right]$ to both sides of (F.1). Using the bounds in (D.2) and (F.2), the drift-plus-penalty term becomes bounded by

$$\Delta\left(\mathbf{U}(k)\right) + V\sum_{i=1}^{N}\mathbb{E}_{\mathbf{U}(k)}\left[r_i(k)T_k\right] \leq C + \mathbb{E}_{\mathbf{U}(k)}\left[T_k\right]\chi(k), \tag{F.3}$$

$$\text{where}\quad \chi(k) \triangleq \sum_{i=1}^{N}\left(V - Y_i(k)\lambda_i\right) r_i(k) + \phi \tag{F.4}$$

$$\text{with}\quad \phi \triangleq \sum_{l=1}^{N}\left(Y_{\pi_l}(k)\lambda_{\pi_l}\,\mathbb{E}_{\mathbf{U}(k)}\left[W_{\pi_l}^{(j)}\right] + X(k)\left(\frac{\mathbb{E}_{\mathbf{U}(k)}\left[\sum_{t\in\mathcal{F}(k)} P_{\pi_l}^{(t)} g_{\pi_l}^{(t)}\right]}{\mathbb{E}_{\mathbf{U}(k)}\left[T_k\right]} - I_{\text{avg}}\right)\right), \tag{F.5}$$

We define the *DOAC* policy to be the policy that jointly finds $\mathbf{r}(k)$, $\{\mathbf{P}^{(t)}\}$ and $\boldsymbol{\pi}(k)$ that minimize $\chi(k)$ subject to the instantaneous interference, the maximum power and the single-SU-per-time-slot constraints in problem (3.8). Step 5-a in the *DOAC* policy minimizes the first summation of $\chi(k)$. For $\{\mathbf{P}^{(t)}\}$ and $\boldsymbol{\pi}(k)$, we can see that ϕ is the only term in the right side of equation (F.4) that is a function of the power allocation policy $\{\mathbf{P}^{(t)}\}$, $\forall t \in \mathcal{F}(k)$. For a fixed priority list $\boldsymbol{\pi}(k)$, using the Lagrange optimization to find the optimum power allocation policy that minimizes ϕ subject to the aforementioned constraints yields equation (3.20), where $P_{\pi_j}(k)$, $\forall i \in \mathcal{N}$, is some fixed power parameter that minimizes Ψ subject to the maximum power constraint only. Substituting by equation (3.20) in ϕ and using the bound $\mathbb{E}_{\mathbf{U}(k)}\left[W^{(j)}_{\pi_l}\right] = W_{\pi_l}(P_{\pi_l}(k)) \leq W^{\text{up}}_{\pi_l}(P_{\pi_l}(k))$ we get Ψ that is defined before equation (3.21). Consequently, $\mathbf{P}^*(k)$ and $\boldsymbol{\pi}^*(k)$, the optimum values for $\mathbf{P}(k)$ and $\boldsymbol{\pi}(k)$ respectively, are ones that minimize Ψ as given by Algorithm 2.

Since the optimum policy that solves (3.8) satisfies the interference constraint, i.e. satisfies $\mathbb{E}_{\mathbf{U}(k)}\left[\sum_{t\in\mathcal{F}(k)} P^{(t)}_{\pi_l} g^{(t)}_{\pi_l}\right] \leq \mathbb{E}_{\mathbf{U}(k)}\left[T_k\right] I_{\text{avg}}$, we can evaluate $\chi(k)$ at this optimum policy with a genie-aided knowledge of $r_i(k) = \bar{W}^*_i$ to get $\chi^{\text{opt}} \triangleq V\sum_{i=1}^{N} \bar{W}^*_i$. Replacing $\chi(k)$ with χ^{opt} in the r.h.s. of (F.3) we get the bound

$$\Delta\left(\mathbf{U}(k)\right) + V\sum_{i=1}^{N} \mathbb{E}_{\mathbf{U}(k)}\left[r_i(k)T_k\right] \leq C + \mathbb{E}_{\mathbf{U}(k)}\left[T_k\right] V\sum_{i=1}^{N} \bar{W}^*_i. \tag{F.6}$$

Taking $\mathbb{E}\left[\cdot\right]$ over this inequality, summing over $k = 0, \cdots, K-1$, denoting $X(0) \triangleq \mathbf{Y}_i(0) \triangleq 0$ for all $i \in \mathcal{N}$, and dividing by $V\sum_{k=0}^{K-1}\mathbb{E}\left[T_k\right]$ we get

$$\frac{\mathbb{E}\left[X^2(K)\right]}{\sum_{k=0}^{K-1}\mathbb{E}\left[T_k\right]} + \sum_{i=1}^{N}\frac{\mathbb{E}\left[Y_i^2(K)\right]}{\sum_{k=0}^{K-1}\mathbb{E}\left[T_k\right]} + \sum_{i=1}^{N}\frac{\sum_{k=0}^{K-1}\mathbb{E}\left[r_i(k)T_k\right]}{\sum_{k=0}^{K-1}\mathbb{E}\left[T_k\right]} \leq \frac{CK}{V\sum_{k=0}^{K-1}\mathbb{E}\left[T_k\right]} + \sum_{i=1}^{N}\bar{W}^*_i. \tag{F.7}$$

Similar steps to those in Appendix D can be followed to prove the mean rate stability of $\{X(k)\}_{k=0}^{\infty}$ and $\{Y_i(k)\}_{k=0}^{\infty}$ as well as the bound in Theorem 4, and thus are omitted here. $\square$

APPENDIX G

PROOF OF THEOREM 8

Proof. We divide the proof into two parts. First, we show that the virtual queues are mean rate stable. This proves that constraints (C3) and (C4) are satisfied. Second, through the Lyapunov optimization technique we show that the drift-minus-reward term is within a constant gap from the performance of the optimal, genie-aided algorithm [61, 62].

Mean Rate Stability: According to (4.17), Algorithm 3 minimizes $\Psi(k)$ where the minimization is taken over all possible scheduling and power allocation algorithms including the optimal algorithm that solves (4.8). We define $\Psi^*(k) \triangleq \min \Psi(k)$. Thus we can write

$$\Psi^*(k) \leq \tilde{\Psi}(k) \tag{G.1}$$

where $\tilde{\Psi}(k)$ is the value of $\Psi(k)$ evaluated at the optimal algorithm and is given by

$$\begin{aligned}\tilde{\Psi}(k) \triangleq &\sum_{i \in \mathcal{N}_{\mathrm{R}}} \mathbb{E}_{\mathbf{U}(k)}\left[Y_i(k)\left(\lambda_i q_i - \tilde{\mathbb{1}}_i(k)\right)\right] + X(k)\left(\sum_{i \in \mathcal{N}} \frac{\mathbb{E}_{\mathbf{U}(k)}\left[\tilde{\mu}_i(k)\tilde{P}_i(k)\right]}{T} - P_{\mathrm{avg}}\right) \\ &+ \sum_{i \in \mathcal{N}_{\mathrm{NR}}} Q_i(k)\left(\mathbb{E}_{\mathbf{U}(k)}\left[LR_i^{(\mathrm{opt})} - \tilde{\mu}_i(k)\tilde{R}_i(k)\right]\right).\end{aligned} \tag{G.2}$$

where $\tilde{P}_i(k)$, $\tilde{\mu}_i(k)$, $\tilde{\mathbb{1}}_i(k)$ and $\tilde{R}_i(k)$ satisfy (4.34), (4.35) and (4.36). Taking $\mathbb{E}[\cdot]$ to (G.2), summing over $k = 0 \cdots K-1$, dividing by K, taking the limit as $K \to \infty$ and using (4.34), (4.35) and (4.36) gives

$$\limsup_{K \to \infty} \frac{1}{K} \sum_{k=0}^{K-1} \mathbb{E}\left[\tilde{\Psi}(k)\right] \leq 0 \tag{G.3}$$

Evaluating by Algorithm 3 in (4.12), and taking $\mathbb{E}[\cdot]$ gives

$$\frac{1}{2}\sum_{i \in \mathcal{N}_{\mathrm{R}}} \mathbb{E}\left[Y_i^2(k)\right] + \frac{1}{2}\sum_{i \in \mathcal{N}_{\mathrm{NR}}} \mathbb{E}\left[Q_i^2(k)\right] + \frac{1}{2}\mathbb{E}\left[X^2(k)\right] \leq C_1 + \mathbb{E}\left[\Psi^*(k)\right]. \tag{G.4}$$

Summing over $k = 0 \cdots K-1$, dividing by K then taking the limit as $K \to \infty$ yields

$$\sum_{i \in \mathcal{N}_{\mathrm{NR}}} \limsup_{K \to \infty} \frac{\mathbb{E}\left[Q_i^2(K)\right]}{2K} + \sum_{i \in \mathcal{N}_{\mathrm{R}}} \limsup_{K \to \infty} \frac{\mathbb{E}\left[Y_i^2(K)\right]}{2K} + \limsup_{K \to \infty} \frac{\mathbb{E}\left[X^2(K)\right]}{2K}$$
$$\leq C_1 + \lim_{K \to \infty} \frac{1}{2K} \sum_{k=0}^{K-1} \mathbb{E}\left[\Psi^*(k)\right] \overset{(a)}{\leq} C_1 + \lim_{K \to \infty} \frac{1}{2K} \sum_{k=0}^{K-1} \mathbb{E}\left[\tilde{\Psi}(k)\right] \overset{(b)}{\leq} C_1. \quad \text{(G.5)}$$

where inequalities (a) and (b) in (G.5) follow from (G.1) and (G.3), respectively. To prove the mean rate stability of the sequence $\{Y_i(k)\}_{k=0}^{\infty}$ for any $i \in \mathcal{N}_{\mathrm{NR}}$, we remove the second and third terms in the first line of (G.5) as well as the summation operator from the second term to obtain $\mathbb{E}\left[Y_i^2(K)\right]/K \leq 2C_1 \; \forall i \in \mathcal{N}$. Using Jensen's inequality we note that

$$\frac{\mathbb{E}\left[Y_i(K)\right]}{K} \leq \sqrt{\frac{\mathbb{E}\left[Y_i^2(K)\right]}{K^2}}. \quad \text{(G.6)}$$

Finally, taking the limit when $K \to \infty$ completes the mean rate stability proof. Similarly we can show the mean rate stability of $X(k)$.

Objective Function Optimality: Evaluating the right-hand-side of (4.14) at the optimal policy that has a genie-aided knowledge of the optimum reward $r_i(k) = R_i^{(\mathrm{opt})}$ we get

$$\Delta(k) - B_{\max} \sum_{i \in \mathcal{N}_{\mathrm{NR}}} \mathbb{E}_{\mathbf{U}(k)}\left[L r_i(k)\right] \leq C_1 + \tilde{\Psi}(k) - B_{\max} \sum_{i \in \mathcal{N}_{\mathrm{NR}}} R_i^{(\mathrm{opt})}. \quad \text{(G.7)}$$

Taking the $\mathbb{E}[\cdot]$ and the time average for both sides of (G.7) and using (G.3) yields

$$\limsup_{K \to \infty} \frac{1}{K} \sum_{k=0}^{K-1} \Delta(k) - B_{\max} \limsup_{K \to \infty} \frac{1}{K} \sum_{k=0}^{K-1} \sum_{i \in \mathcal{N}_{\mathrm{NR}}} \mathbb{E}_{\mathbf{U}(k)}\left[L r_i(k)\right] \leq$$
$$C_1 - B_{\max} \sum_{i \in \mathcal{N}_{\mathrm{NR}}} L R_i^{(\mathrm{opt})}. \quad \text{(G.8)}$$

Removing the first term in (G.8) and rearranging gives

$$\limsup_{K \to \infty} \frac{1}{K} \sum_{k=0}^{K-1} \sum_{i \in \mathcal{N}_{\mathrm{NR}}} \mathbb{E}_{\mathbf{U}(k)}\left[r_i(k)\right] \geq -\frac{C_1}{L B_{\max}} + \sum_{i \in \mathcal{N}_{\mathrm{NR}}} R_i^{(\mathrm{opt})}. \quad \text{(G.9)}$$

Using (4.1) and (C2) completes the proof of (4.37). $\square$

APPENDIX H

PROOF OF THEOREM 13

Proof. We divide our proof into two parts. In the first part (Achievability), we show that if $\boldsymbol{\lambda}_{\mathrm{NR}}$ is strictly within the region $\mathcal{R}_{\mathrm{Lamb}}$, then the queues can be stabilized. And the algorithm that stabilizes these queues is a modified version of Algorithm 5. We show this using the Lyapunov optimization technique [60, pp.120]. In the second part (Converse), we show that if $\boldsymbol{\lambda}_{\mathrm{NR}}$ is outside the region $\mathcal{R}_{\mathrm{Lamb}}$, then there exists no algorithm that guarantees the stability of the NRT queues.

Achievability: We will show here that the following inequality holds under Algorithm 5 which is the key to the proof.

$$\sum_{i\in\mathcal{N}_{\mathrm{NR}}} \lambda_i Q_i(k) + \sum_{i\in\mathcal{N}_{\mathrm{R}}} \lambda_i q_i Y_i(k) - \sum_{i\in\mathcal{N}} X(k) P_{\mathrm{avg}} \leq \mathbb{E}_{\mathbf{U}(k)}\left[\sum_{i\in\mathcal{N}_{\mathrm{NR}}} Q_i(k) D_i(k) + \sum_{i\in\mathcal{N}_{\mathrm{R}}} Y_i(k) D_i(k) - \sum_{i\in\mathcal{N}} \frac{X(k)\mu_i\left(k\right)P_i\left(k\right)}{T}\right], \quad \text{(H.1)}$$

where $B_i(k) \triangleq \mu_i\left(k\right)\log\left(1+P_i\left(k\right)\gamma_i(k)\right)$. Once this inequality is proven, the rest of the achievability proof works similar to Theorem 5.3.2 in [60, pp.120].

Since $\boldsymbol{\lambda}_{\mathrm{NR}}(1+\epsilon) \in \mathcal{R}_{\mathrm{Lamb}}$, to prove (H.1) we multiply (4.49) by λ_i, (4.50) by λ_i, and (4.52) by $(-P_{\mathrm{avg}})$, then add the three inequalities after summing the first over $i \in \mathcal{N}_{\mathrm{NR}}$ and the second over $i \in \mathcal{N}_{\mathrm{R}}$ yielding

$$\sum_{i\in\mathcal{N}_{\mathrm{NR}}} \lambda_i Q_i(k) + \sum_{i\in\mathcal{N}_{\mathrm{R}}} \lambda_i q_i Y_i(k) - \sum_{i\in\mathcal{N}} X(k) P_{\mathrm{avg}} \leq \sum_{\boldsymbol{m}\in\mathcal{M}^N}\left(\sum_{i\in\mathcal{N}_{\mathrm{NR}}} Q_i(k) D_i(\boldsymbol{m},k) + \sum_{i\in\mathcal{N}_{\mathrm{R}}} Y_i(k) D_i(\boldsymbol{m},k) - \sum_{i\in\mathcal{N}} \frac{X(k)\mu_i\left(\boldsymbol{m},k\right)P_i\left(\boldsymbol{m},k\right)}{T}\right)\pi_{\mathbf{m}} \quad \text{(H.2)}$$

$$\leq \sum_{\boldsymbol{m}\in\mathcal{M}^N}\left[\sum_{i\in\mathcal{N}_{\mathrm{NR}}} \Psi^*_{\mathrm{NR}}(i,k) + \sum_{i\in\mathcal{N}_{\mathrm{R}}} \Psi^*_{\mathrm{R}}(i,k)\right]\pi_{\mathbf{m}}, \quad \text{(H.3)}$$

where $D_i(\boldsymbol{m},k) \triangleq \mu_i\left(k\right)\log\left(1+P_i\left(k\right)\gamma_i(k)\right)$ while inequality (H.2) follows since the objective of problem (4.17) is an upper bound on (H.2). But since the right-hand-side of (H.1) can be manipulated to give

$$\mathbb{E}_{\mathbf{U}(k)}\left[\sum_{i\in\mathcal{N}_{\mathrm{NR}}} Q_i(k)D_i(\boldsymbol{m},k)+\sum_{i\in\mathcal{N}_{\mathrm{R}}} Y_i(k)D_i(\boldsymbol{m},k)-\sum_{i\in\mathcal{N}}\frac{X(k)\mu_i\left(k\right)P_i\left(k\right)}{T}\right]$$

$$=\sum_{i\in\mathcal{N}_{\mathrm{NR}}}\mathbb{E}_{\mathbf{U}(k)}\left[Q_i(k)D_i(\boldsymbol{m},k)-\frac{X(k)\mu_i\left(k\right)P_i\left(k\right)}{T}\right]$$

$$+\sum_{i\in\mathcal{N}_{\mathrm{R}}}\mathbb{E}_{\mathbf{U}(k)}\left[Y_i(k)D_i(\boldsymbol{m},k)-\frac{X(k)\mu_i\left(k\right)P_i\left(k\right)}{T}\right] \tag{H.4}$$

$$=\sum_{\boldsymbol{m}\in\mathcal{M}^N}\left[\sum_{i\in\mathcal{N}_{\mathrm{NR}}}\Psi^*_{\mathrm{NR}}(i,k)+\sum_{i\in\mathcal{N}_{\mathrm{R}}}\Psi^*_{\mathrm{R}}(i,k)\right]\pi_{\mathbf{m}} \tag{H.5}$$

$$\geq\sum_{i\in\mathcal{N}_{\mathrm{NR}}}\lambda_iQ_i(k)+\sum_{i\in\mathcal{N}_{\mathrm{R}}}\lambda_iq_iY_i(k)-\sum_{i\in\mathcal{N}}X(k)P_{\mathrm{avg}} \tag{H.6}$$

where (H.5) follows by evaluating (H.4) at Algorithm 5 while (H.6) follows from (H.3) which completes the proof of (H.1).

Converse: The converse is done by showing that the upper bound of the sum of the number bits served from all NRT buffers under the best, possibly genie-aided, policy is less than the sum of bits arriving to the NRT buffers if the arrival rate does not satisfy (4.50) through (4.54).

From the strict separation theorem [60, pp.10], if $\lambda\notin\mathcal{R}_{\mathrm{Lamb}}$ then there exists a vector $\beta\triangleq[\beta_1,\cdots\beta_{N_{\mathrm{NR}}}]^{\mathrm{T}}\in\mathcal{R}^{N_{\mathrm{NR}}}$ and a constant $\delta>0$ such that for any vector $x\in\mathcal{R}_{\mathrm{Lamb}}$ the following holds

$$\sum_{i\in\mathcal{N}_{\mathrm{NR}}}\beta_i\lambda_i\geq\sum_{i\in\mathcal{N}_{\mathrm{NR}}}\beta_ix_i+\delta \tag{H.7}$$

Define $H(k+1)=H(k)+\sum_{i\in\mathcal{N}_{\mathrm{NR}}}\beta_i\left(La_i(k)-B_i(k)\right)$ as the weighted sum of the queues where $B_i(k)\triangleq\mu_i\left(k\right)R_i(k)$ is the number of bits transmitted to user i at slot k. Hence we have

$$H(K)=\sum_{k=0}^{K-1}\sum_{i\in\mathcal{N}_{\mathrm{NR}}}\beta_i\left(La_i(k)-B_i(k)\right). \tag{H.8}$$

Define the set $\mathcal{K}_K(l)\triangleq\{k:m(k)=l,0\leq k<K\}$ we can bound the second term in

(H.8) as follows

$$\sum_{i\in\mathcal{N}_{\mathrm{NR}}}\beta_i \limsup_{K\to\infty}\sum_{k=0}^{K-1}\frac{B_i(k)}{K} \le \sum_{i\in\mathcal{N}_{\mathrm{NR}}}\beta_i \limsup_{K\to\infty}\sum_{l=1}^{M}\sum_{k\in\mathcal{K}_K(l)}\frac{\tilde{B}_i(k)}{|\mathcal{K}_K(l)|}\frac{|\mathcal{K}_K(l)|}{K} \tag{H.9}$$

$$= \sum_{i\in\mathcal{N}_{\mathrm{NR}}}\beta_i \sum_{l=1}^{M}\tilde{B}_i^{(l)}\pi_l = \sum_{i\in\mathcal{N}_{\mathrm{NR}}}\beta_i \sum_{l=1}^{M} Lx_i\pi_l. \tag{H.10}$$

Adding $L\delta$ to both sides of (H.10) and using (H.7) yields

$$\sum_{i\in\mathcal{N}_{\mathrm{NR}}}\beta_i \limsup_{K\to\infty}\sum_{k=0}^{K-1}\frac{B_i(k)}{K} + L\delta \le L\left(\sum_{l=1}^{M}\pi_l\sum_{i\in\mathcal{N}_{\mathrm{NR}}}\beta_i x_i + \delta\right) \le \sum_{i\in\mathcal{N}_{\mathrm{NR}}}\beta_i L\lambda_i$$

$$= \lim_{K\to\infty}\sum_{k=0}^{K-1}\frac{La_i(k)}{K}. \tag{H.11}$$

Combining (H.11) and (H.8) we conclude that $\limsup_{K\to\infty} H(K) = \infty$ which means that the weighted sum of the queues is unbounded, under the best possible policy, when $\boldsymbol{\lambda}_{\mathrm{NR}} \notin \mathcal{R}_{\mathrm{Lamb}}$ which completes the proof. $\square$