Scheduling and Resource Allocation in Multi-user Wireless Systems

by

Xuan Wang
B.Eng., Beijing University of Posts and Telecommunications, 2007
M.S., Beijing University of Posts and Telecommunications, 2010

A Dissertation Submitted in Partial Fulfillment of the
Requirements for the Degree of

DOCTOR OF PHILOSOPHY

in the Department of Electrical and Computer Engineering

Scheduling and Resource Allocation in Multi-user Wireless Systems

by

Xuan Wang
B.Eng., Beijing University of Posts and Telecommunications, 2007
M.S., Beijing University of Posts and Telecommunications, 2010

Supervisory Committee

Dr. L. Cai, Supervisor
(Department of Electrical and Computer Engineering)

Dr. H.C. Yang, Departmental Member
(Department of Electrical and Computer Engineering)

Dr. K. Wu, Outside Member
(Department of Computer Science)

Supervisory Committee

Dr. L. Cai, Supervisor
(Department of Electrical and Computer Engineering)

Dr. H.C. Yang, Departmental Member
(Department of Electrical and Computer Engineering)

Dr. K. Wu, Outside Member
(Department of Computer Science)

ABSTRACT

In this dissertation, we discuss various aspects of scheduling and resource allocation in multi-user wireless systems.

This work starts from how to utilize advanced physical-layer technology to improve the system performance in a multi-user environment. We show that by using superposition coding (SPC) and successive interference cancellation, the system performance can be greatly improved with utility-based scheduling. Several observations are made as the design guideline for such system. Scheduling algorithms are designed for a system with hierarchical modulation which is a practical implementation of SPC.

However, when the utility-based scheduling is designed, it is based on the assumption that the system is saturated, *i.e.*, users in the system always have data to transmit. It is pointed out in the literature that in a system with stochastic traffic, even if the arrival rate lies inside the capacity region, the system in terms of queue might not be stable with the utility-based scheduling. Motivated by this, we have studied the stability region of a general utility-based scheduling in a multi-user system with stochastic traffic. We show that the stability region is generally less than the capacity region, depends on how to interpret an intermediate control variable, and the resultant stability region may be even non-convex and exhibits undesirable properties which should be avoided.

As the utility-based scheduling cannot achieve throughput-optimal, we turn our attentions to the throughput-optimal scheduling algorithms, whose stability region is identical to the capacity region. The limiting properties of an overloaded wireless system with throughput-optimal scheduling algorithms are studied. The results show that the queue length is unstable however the scheduling function of the queue length is stable, and the average throughput of the system converges.

Finally we study how to schedule users in a multi-user wireless system with information-theoretic security support, which is focused on the secrecy outage probability. The problem is essentially about how to schedule users, and allocate resources to stabilize the system and minimize the secrecy outage probability. We show that there is a tradeoff between the arrival rate of the traffic and the secrecy outage probability. The relative channel condition of the eavesdropper also plays an important role to the secrecy outage probability.

In summary, we showed utility-based scheduling using SPC can improve the system performance greatly, but the utility-based scheduling has limitations: the stability region might not have desired properties. On the contrary throughput-optimal scheduling has its own drawbacks: the traffic cannot be handled properly if the system is overloaded. The further study on the secrecy outage probability gives guideline on how to design a scheduler in a system with information-theoretic security support.

Contents

List of Tables

List of Figures

List of Acronyms

AWGN ..... Additive White Gaussian Noise

AQM ..... active queue management

BER ..... Bit Error Rate

BLER ..... Block Error Rate

CSI ..... Channel State Information

CRB ..... Channel Rate Based

DVB ..... Digital Video Broadcasting

FIFO ..... First-in-first-out

GBC ..... Gaussian Broadcast Channel

HM ..... Hierarchical Modulation

MAC ..... Media Access Control

OFDM ..... Orthogonal Frequency-Division Multiplexing

PF ..... Proportionally Fair

QoS ..... Quality of Service

SIC ..... Successive Interference Cancellation

SNR ..... Signal-to-noise Ratio

SPC ..... Superposition Coding

TDMA ..... Time Division Multiple Access

UB ..... Utility Based

ACKNOWLEDGEMENTS

Foremost, I would like to express my sincere gratitude to supervisor Prof. Lin Cai for her support of my research and study at University of Victoria in the past four years, for her patience, inspiration and technical advice. I could not have imagined having a better advisor and mentor for my Ph.D study.

Besides my advisor, I would like to thank Prof. Hong–Chuan Yang and Prof. Kui Wu for serving as my thesis committee, Prof. Ye Xia from University of Florida to serve as my external examiner.

My sincere thanks also goes to Prof. Jianping Pan for his valuable comments, insights and guidance.

I thank my fellow labmates in Communication Networks Lab: Dr. Zhe Yang, Dr. Yuanqian Luo, Dr. Siyuan Xiang, Lei Zheng, Min Xing, Kan Zhou, Yi Chen, Zhe Wei and Haoyuan Zhang for the stimulating discussions and for the funs we have in the last four years. Also I thank my friends at Beijing University of Posts and Telecommunications: Dr. Chao Dong, Chi Liu and Jinan Ma, for your encouragement.

Last and certainly not least, I would like to thank my parents, for their endless love and supporting me spiritually throughout my life.

Xuan Wang, Burnaby, BC, Canada

DEDICATION

To my parents,
And all of my friends,
Without whom none of this would be possible.

Chapter 1

Introduction

In this dissertation, we discuss various aspects of resource allocation in multi-user wireless systems. Scheduling algorithms in a saturated system using superposition coding/hierarchical modulation are designed and the remarkable performance improvement is demonstrated. The stability regions of utility-based opportunistic scheduling algorithms in a system with stochastic traffic are derived and the structure properties are obtained. The limiting properties of an overloaded system with throughput-optimal scheduling algorithm are quantified and the corresponding throughput is analyzed. Secrecy outage probability in a multi-user wireless system has been investigated through a resource allocation problem and two optimal algorithms, one online and one offline, are proposed to solve the resource allocation problem.

1.1 Background

Since the available resources in a wireless network are limited, and users are competing for the limited resources, how to allocate the resources to users fairly and efficiently is one of the key problems in the operation of a wireless system.

In the literatures there are different assumptions on the system. In this dissertation, we only consider a multi-user wireless system with one base station and N users. Under this general assumption, the study of scheduling and resource allocation in multi-user wireless systems has two distinct origins.

One is originated from the information-theoretic point of view. This research aims to quantify the capacity region of the channel underlying the multi-user wireless system. For example, with proper physical-layer technology, the equivalent channel

of multi-carrier wireless systems can be modelled as a parallel Gaussian broadcast channel (GBC), whose capacity region of parallel Gaussian broadcast channel was studied in [82, 83]. In order to achieve the capacity region of parallel GBC, the corresponding power allocation scheme was developed in [82, 83]. As the resource allocation scheme aims to achieve the capacity region, the system is assumed to be saturated, *i.e.*, every user in the system always has data to transmit.

However, as the capacity region of wireless channel typically cannot be achieved in a practical wireless system, the capacity region can be replaced by achievable rate region which is determined by the practical constraint of wireless systems. For example, the downlink scheduling in an OFDMA wireless system was reviewed in [70]. Although this kind of research focus on a specific wireless system with certain physical-layer technologies (such as OFDM technology with adaptive modulation and coding), typically they assume that the system is saturated, which is aligning with the research on the capacity region.

With the algorithms to achieve the capacity region or achievable region of a multiuser wireless system, it is natural to have an objective to quantify the algorithms. As the resource allocation is about the rate allocated to users, usually a function of the rates allocated to users is the objective of the resource allocation, and often referred as utility. The utility-based resource allocation in wireless system is reviewed in [96] with a unified framework considering different quality-of-service requirements.

The other is originated from the study of queueing network and stability. For a queueing system, the arrival traffic rate should be smaller than the service rate in order to stabilize the system (queues in the system). Correspondingly, in a wireless system, the sufficient condition to stabilize the system is that the arrival rate lies inside the capacity region. However, with the sufficient condition, not all scheduling algorithm can lead to a stable system. In [80] the author showed that there is one algorithm that can stabilize the wireless system with ON-OFF channels. Typically, if an algorithm can stabilize the system if possible, it is referred as a throughput-optimal scheduling algorithm. Motivated by [80], different kinds of throughput-optimal scheduling were proposed, and were generalized in [12].

1.2 Motivation

Since the utility-based scheduling algorithm aims to maximizing the utility of the (saturated) system, and generally the utility is a function of the throughput, which

should lie inside the capacity region, if the capacity region can be improved by using advanced physical-layer technology without increasing the system overhead (such as signaling), the utility of the system can be improved.

Since the system may not be always saturated, the traffic for each user can be modeled as a stochastic process. As a result, it is important to understand whether the queues in the system can be stabilized for a given traffic arrival rate. It is already known that the utility-based scheduling algorithm cannot provide the maximal stability region, and thus it is important to quantify the stability region of the utility-based scheduling algorithm.

In the literature, the study for throughput-optimal scheduling is generally focusing on the scenario that the system is underloaded, or is able to be stabilized. The system behavior for the overloaded system is not fully understood. Moreover, only the throughput aspect of the system is studied in this area, and some other aspects, such as security, is not studied for throughput-optimal scheduling.

These open issues motivate this dissertation.

1.3 Research Objectives and Contributions

This dissertation has made several contributions: designed scheduling algorithms aiming to improve the multi-user system performance by using advanced physical-layer technologies – SPC and HM; derived the stability region of the utility-based scheduling algorithm in a system with stochastic traffic; obtained the limiting properties of an overloaded system with throughput-optimal scheduling algorithms; designed scheduling algorithms that consider the security issue in a system with information-theoretic encoder/decoder. The detailed research objects and contributions are discussed as follows.

1.3.1 Scheduling in SPC/HM-Aided Wireless System

The most fundamental resource in a wireless network is the physical spectrum of the wireless channel, which is limited. In order to provide high data-rate services for users, one of the main research objectives is maximizing the spectrum efficiency, which is a typical objective of the research on the physical layer, *i.e.*, increasing the resource availability of a single point-to-point link.

While the practical system is typically a multi-user system, and recent research

shows that the performance of a multi-user wireless system can be improved by efficient scheduling algorithms [83, 85]. This motivates the research on the opportunistic scheduling, which tries to utilize the fluctuation of the channel to improve the system performance, mostly to increase the system throughput.

A multi-user wireless system also means the downlink channel is a broadcast channel. It is well-known that the capacity region of a degraded Gaussian broadcast channel can be achieved by using superposition coding (SPC) and successive interference cancellation (SIC) technology.

As both SPC and opportunistic scheduling need the instantaneous channel state information (CSI), by utilizing these two technologies together, the system performance can be improved. Motivated by this, in this work, we have designed utility-based opportunistic scheduling algorithms in a SPC-aided multi-user wireless system, try to improve the system utility. Moreover, we have investigated the performance gain introduced by using SPC and SIC, and further discussed the scheduling and resource allocation algorithm design in a system where hierarchical modulation (HM) is used as a practical implementation of SPC.

1.3.2 Stability Region of Opportunistic Scheduling in Wireless Systems

Traditionally, the scheduler in a wireless network is designed based on the assumption that the system is saturated, and the number of users in the system is a constant. The assumption simplifies the problem, but as shown in [2], these kinds of schedulers may lead the unsaturated system to be unstable, while the system in the same circumstance can be stabilized by other scheduling policies, such as max-weight scheduling [80].

There is little work done to quantify the stability region of opportunistic scheduling. The stability region of an opportunistic scheduling policy in a two-user wireless network with i.i.d. Bernoulli arrival traffic was derived in [23]. In [67], the authors discussed the two-user stability region in a static channel configuration with concurrent transmissions. As the above two works only considered some special scheduling algorithms to study the stability in a two-user wireless system, they are not general enough to observe the general properties of the stability region of utility-based scheduling algorithms, which motivates us to study the stability region of utility-based scheduling in a system with stochastic traffic.

In this work, we have studied the stability regions of two utility-based opportunis-

tic scheduling policies: the utility-based (UB) scheduling and the channel-rate-based (CRB) scheduling, with a general traffic arrival in a wireless system with N users. For the UB scheduling, the explicit closed-form stability region generally cannot be obtained, while we develop a theorem to examine the stability of a system given the arrival rate, and a numerical method is provided to obtain the stability region in a two-user system. We have further studied the properties of the stability region of the UB scheduling, and showed that it is generally non-convex and may also exhibit some undesirable features. For instance, decreasing the arrival rate of one user may lead the system to be unstable. For the CRB scheduling, we have obtained the closed-form expression of the stability region, which is a convex hull. Besides the stability region, we have further studied the extended stability region by giving a weight to each user. The results show that by varying the weight assigned to each user, the union of the resultant stability region is equal to the ergodic capacity region, for both scheduling policies. This suggests as long as the system can be stabilized, by assigning a proper weight to each user, using a non-throughput-optimal scheduling may also stabilize the system.

1.3.3 Overloaded Wireless System Performance

If the resource of the system is sufficient to fulfill the demand of users and maintain the stability of the system, then there should have a resource allocation scheme to do so, which is usually called throughput-optimal scheduling. This kind of scheduling algorithms should consider the incoming traffic, and thus needs more knowledge compared with the resource allocation schemes for a saturated system.

The performance of such throughput-optimal scheduling has been extensively investigated under the assumption that the system is stable, or underloaded. However, it is inevitable that a system may experience overloaded periods in practice due to the fluctuation of the traffic volume [6]. Therefore, it is important to study the system performance with throughput-optimal scheduling algorithms if the system is overloaded. The state-of-the-art research in this area has concluded only for some special throughput-optimal scheduling policies, such as MaxWeight scheduling in [9] and general-MaxWeight scheduling in [72]. The general system behavior of an overloaded system is still missing.

In this work, we have studied the limiting properties of overloaded multiuser wireless systems with infinite buffer and a general throughput-optimal scheduling

policy, quantified the network performance of two special and widely used throughput-optimal scheduling algorithms. Furthermore, we have analyzed the performance of a finite-buffer system with Drop-Tail queue [14] and various buffer-sharing schemes, which is of practical interests and often missed in the literature.

The results show that the although the system is overloaded, the scheduling function of queue length converges with the infinite-buffer assumption and the average throughput converges and can be obtained by solving a convex optimization problem. With finite buffer assumption, the system performance is highly related with the buffer scheme and exhibit complicated relationship.

1.3.4 Secrecy Outage Probability in Multiuser Wireless Systems

More recently, information-theoretic security has been widely discussed as it quantified the fundamental system secrecy. How to allocate the resource to achieve certain secrecy is an important issue. However, most of the works are discussed from a traditional information-theoretical perspective, *i.e.*, quantifying the capacity region under different network settings. All these works [98, 45, 21, 36, 62] tried to solve an optimization problem, implicitly or explicitly, based on the assumption that the system is saturated and each user in the system always has data to transmit. Only the reliability and security issues are considered, and the stability issue is ignored since it is typically treated in the higher layer. However, the stability is of equal importance with reliability and security, since it further determines whether a practical system can work properly and desirably over a sufficiently long time period.

Motivated by this, we studied the scheduling problem in multiuser wireless systems, where one eavesdropper exists in the system. We considered minimizing the secrecy outage probability of the system, which is a coding-delay-limited metric that is of practical interests. Besides, we further considered the queue stability issue which is often ignored in the work that maximizes the ergodic achievable rate. Therefore, the scheduling problem was formulated as an optimization problem minimizing the system secrecy outage probability (security issue) and subject to the constraints that the queues in the system should be stable (stability issue) and the transmission rate does not exceed the capacity region (reliability issue).

Little work has been done jointly considering these three aspects. Some works assumed that the eavesdroppers' channel state information at symbol level (full in-

stantaneous CSI) can be obtained by the BS, such as [50, 22, 54], which may not be practical. Some works, such as [64], relax the assumption on the instantaneous CSI, however, the designed scheme is not scalable to a case with multiple legitimate receivers, which limits the usage of the proposed algorithm.

In this work, we have discussed the secrecy outage probability in a multi-user wireless system with stochastic traffic and channel-adaptive transmission, designed a scalable scheduling algorithm with a weak assumption that only the distribution of the CSI of the eavesdropper is known by the BS, and further showed that directly applying the well-know Lyapunov optimization framework to the formulated optimization problem cannot lead to the optimal solution, as the queue length is not always a proper "online representation" of Lagrangian multiplier.

1.4 Dissertation Organization

This work focuses on the scheduling algorithm design and analysis in a multi-user wireless system. The rest of this dissertation is organized as follows.

In Chapter 2, we discuss the resource allocation problem in an SPC-aided wireless network. The resource allocation problem is formulated and several algorithms having different computational complexities are proposed. Through simulations we study the performance gain achieved by SPC and make several observations that can be used as a guideline for the system design.

In Chapter 3, we discuss the scheduling algorithms in a two-layer HM-aided wireless network. We formulate the scheduling problem and propose two algorithms with different computational complexities. The simulation demonstrates that the propose algorithm can achieve significant performance improvement.

Chapter 4 discusses the stability region of two opportunistic scheduling algorithms, the CRB algorithm and the UB algorithm, that were originally designed for a saturated wireless system. The results show that the stability regions generally are both smaller than the capacity region, and that of UB may even be non-convex which may lead to certain undesired property that should be avoided in a practical system.

The system performance in an overloaded wireless system with throughput-optimal scheduling algorithm is discussed in Chapter 5. We show that in such system setting generally all the queues in the system are unstable, but the average throughput converges. We further discuss how to obtain the average throughput, which can be used to analyze the system performance in a temporarily overloaded system.

The scheduling algorithm in a secrecy wireless system is presented in Chapter 6. We extend the secrecy outage probability for a system with channel-adaptive transmission. We discuss how to minimize the secrecy outage probability subject to reliability and stability constraint. Simulation results show that there is a tradeoff between the arrival rate and the secrecy outage probability, and the relative channel condition also has a great impact on the secrecy outage probability.

Chapter 7 concludes this dissertation.

In the rest of this dissertation, bold face letters represent vectors and calligraphic letters represent sets.

1.5 Bibliographic Notes

Most of the works reported in this dissertation have appeared in research papers. The works in Chapter 2 have been published in [90]. The works in Chapter 3 have been published in [89]. The works in Chapter 4 have been published in [91]. The works in Chapter 5 have been published in [88] and those in Chapter 6 have appeared in [94], and been submitted as [93].

Chapter 2

Resource Allocation in a K-User Wireless Broadcast System with N-Layer Superposition Coding

Theoretically, SPC can achieve the capacity of a degraded Gaussian broadcast channel. In this chapter, we study the resource allocation problem in a K-user wireless broadcast system with N-layer SPC. The problem is formulated as a sum-utility maximization problem based on the average throughput, and three algorithms are proposed to solve the problem. The simulation results show that the SPC gain highly depends on the variability of the channel and the SNR range of channels for different users. SPC is more favourable in the scenario with small-variation fast-fading channel and a wide SNR range of channels for different users.

2.1 Introduction and Related Work

It is well known that the capacity of a broadcast channel generally cannot be achieved by an orthogonal resource allocation, such as time division multiple access (TDMA). To achieve the capacity region of a broadcast channel, various techniques were proposed for different scenarios. The simplest scenario is the Gaussian broadcast channel (GBC), the typical channel model for many wireless communication systems, such as single-antenna systems and zero-forcing MIMO systems. To achieve the capacity of a GBC channel, the signals of different users are superimposed in the ascending order of the received signal-to-noise ratio (SNR), and the receiver uses the successive

interference cancellation to extract the useful signal. So long as the SNR of different user is not identical, this SPC approach can result in an extra capacity region, compared to the orthogonal resource allocations. In a multi-user wireless system, since the distance between the user and the base station is varying, using SPC can result in the extra capacity region and an enhanced system performance. Considering this feature of SPC, various works have been done in the resource allocation area.

[82] studied the optimal power allocation to achieve the capacity region in a parallel GBC. The parallel GBC is a more general GBC and the results are also applicable to the GBC. From [82], the boundary of the capacity region can be achieved by solving a weighted-sum-rate maximization problem, and an optimal algorithm was proposed. This work was further extended in [97] by giving a minimum rate constraint. In [61], a dual decomposition method was used to build an optimization framework to a general resource allocation problem in a parallel GBC. These work assumed that the capacity region of the parallel GBC is achievable, which means all the signals of users can be superimposed together. In practical, due to the high complexity to decode the multi-user signal, the number of signals to be superimposed should be limited, and thus the above results may not be applicable. A more practical problem is discussed in [1], where only two-layer SPC is used. With the proportional fairness constraint, a guideline about how to select the user group was proposed. This work only considered a specific resource allocation objective, and it is not extensible to multi-user cases.

In this work, we consider a more general objective of resource allocation. The sum-utility of the users received in the long term is maximized, which includes the weighted-sum-rate maximization and the proportional fairness maximization as two special cases. We also use a more practical assumption of the SPC signal. We assume that, in a K-user system, upto N-layer SPC can be used. This assumption is general, since by varying N, all the possible settings are included. The sum-utility in the long term is based on the average throughput, and thus it cannot be solved directly. Based on the stochastic approximation, solving an approximated problem iteratively can reach the optimality. Using the primal decomposition, the approximated problem can be decomposed into a user group scheduling problem and a weighted-sum-rate maximization problem. By solving the two problems jointly yields the optimal solution which has a high computational complexity.

The main contributions of this chapter are three-fold. First, we formulate a general SPC resource allocation problem, where the number of layers of SPC is arbitrary and it may or may not be identical to the number of users. Second, we not only consider

the optimal solution, which is of high computational complexity, but also propose several low-complexity and low-overhead solutions, which are more practical. Third, extensive simulation results are presented, which show the benefit of using SPC, and the tradeoff between performance and complexity. The results provide a practical guideline for system design.

2.2 System Model

2.2.1 Fading Broadcast Channel

We consider a K-user block fading broadcast channel, where the channel gain is a constant within each block, and experiences independent and identical fading during blocks. The length of the block is identical. For different users, the statistic properties of the fading channel are not necessarily the same. The noise of each user is assumed to be an additive white Gaussian noise. The transmitter and the receiver both can track the channel and have the channel state information (CSI). There is a peak power constraint P in each fading block at the transmitter.

In the fading block t, the received signal of user i is $y_i(t) = h_i(t)x_i(t)+z_i(t)$, for $i = 1, 2, ..., K$, where $h_i(t)$ is the channel gain of user i, $x_i(t)$ is the transmitted signal of user i and $z_i(t)$ is the zero-mean complex white Gaussian noise of user i, with power N_0W. The channel gain can be normalized into the noise term, so the equivalent received signal is given by

$$\hat{y}_i(t) = x_i(t) + \hat{z}_i(t), \quad i = 1, 2, ..., K, \tag{2.1}$$

where $\hat{z}_i(t) = z_i(t)/h_i(t)$ with power $n_i(t) = N_0W/|h_i(t)|^2$.

The SNR can be calculated by

$$\gamma_i(t) = P|h_i(t)|^2/N_0W, \quad i = 1, 2, ..., K.$$

2.2.2 Achievable Rate Region

In each fading block t, the channel is a K-user GBC, whose capacity can be achieved by a K-layer SPC. Since we assume only N-layer SPC is used, the capacity generally is not achievable. In this subsection, we will obtain the achievable rate region of the considered system in each fading block. In the following, all the fading block indexes

are omitted.

We use $\mathcal{K}$ to denote the set of all users and $\mathcal{N}(k), k = 1, 2, ..., \frac{K!}{N!(K-N)!}$ to denote the k-th N-user group. Denote the power set of $\mathcal{N}(k)$ as $\mathcal{S}$, so $|S| = \frac{K!}{N!(K-N)!}$.

Using the signal model in (2.1), and assuming the general order $n_1 \leq n_2 \leq ... \leq n_K$, based on the capacity region of the N-user degraded GBC [19], the achievable rate of K users when selecting user group $\mathcal{N}(k)$ is denoted as

$$\begin{aligned}
&\mathcal{C}_{\mathcal{K}}^{\mathcal{N}(k)} = \{\mathbf{r} : \\
&r_i \leq \log_2(1 + \frac{\alpha_i P}{n_i + \sum_{j<i} \alpha_j P}) \quad i = 1, 2, ..., K, \\
&\sum_i \alpha_i = 1 \ \text{ and } \alpha_i = 0 \text{ if } i \notin \mathcal{N}(k)\},
\end{aligned}$$

where α_i is the fraction of power allocated to user i.

The achievable rate region of the K-user GBC with N-layer SPC is the convex hull of $\mathcal{C}_{\mathcal{K}}^{\mathcal{N}(k)}$, which can be obtained as

$$\mathcal{C}_{\mathcal{K}} = \bigcup_{\sum t_k = 1} \sum_{k=1}^{|S|} \mathcal{C}_{\mathcal{K}}^{\mathcal{N}(k)} t_k,$$

where $t_k \in \mathbb{R}_+ \bigcup \{0\}$ is the time sharing factor of the k-th N-user group.

Then the average achievable rate region of K users $\bar{\mathcal{C}}_{\mathcal{K}}$ is the convex hull of all $\mathcal{C}_{\mathcal{K}}$ of each fading block.

2.3 Problem Formulation

2.3.1 Resource Allocation Problem

The objective of resource allocation is to maximize the sum-utility of all users in the long term, where the utility is defined as a function of the average throughput. The average rates allocated to users can be obtained by

$$\mathbf{R}_{\mathcal{K}} = \arg\max_{\boldsymbol{\eta} \in \bar{\mathcal{C}}_{\mathcal{K}}} \sum_{i \in \mathcal{K}} U(\eta_i), \tag{2.2}$$

where $U(x)$ is the utility function which is assumed to be concave, monotonically non-decreasing and differentiable.

According to [96], solving the first-order approximation of (2.2) in each fading block can solve the problem, and the convergence is guaranteed by stochastic approximation [44]. Thus, the online scheduling algorithm in fading block t is

$$\mathbf{r}_{\mathcal{K}}(t) = \arg\max_{\boldsymbol{\eta}\in\mathcal{C}_{\mathcal{K}}(t)} \sum_{i\in\mathcal{K}} U'(R_i(t))(\eta_i - R_i(t)), \tag{2.3}$$

where $\mathbf{r}_{\mathcal{K}}(t)$ is the allocated rate of user group $\mathcal{K}$ in fading block t, $\mathcal{C}_{\mathcal{K}}(t)$ is the achievable rate region of $\mathcal{K}$ in block t, and $\mathbf{R}(t)$ is the measured average throughputs before t. $\mathbf{R}(t)$ is updated by $\mathbf{R}(t) = \mathbf{R}(t-1)+\epsilon(\mathbf{r}(t-1)-\mathbf{R}(t-1))$, where ϵ is the step size used to control the convergence speed and accuracy. By removing the constant term in (2.3) to simplify the objective function, the online scheduling algorithm in each fading block can be rewritten as

$$\mathbf{r}_{\mathcal{K}} = \arg\max_{\boldsymbol{\eta}\in\mathcal{C}_{\mathcal{K}}} \sum_{i\in\mathcal{K}} U'(R_i)\eta_i, \tag{2.4}$$

where the fading block index is omitted.

The problem cannot be directly solved, since the achievable rate region $\mathcal{C}_{\mathcal{K}}$ is not likely to be written in an explicit closed form. But since the constraint set is a closed convex hull, (2.4) can be reformulated as

$$\mathbf{r}_{\mathcal{K}} = \arg\max_{\substack{\sum t_k=1 \\ \boldsymbol{\eta}\in\mathcal{C}_{\mathcal{K}}^{\mathcal{N}(k)}}} \sum_k t_k \sum_{i\in\mathcal{K}} U'(R_i)\eta_i,$$

which can be further decomposed into two sub-problems by primal decomposition. The first is a weighted-sum-rate maximization problem for every user group

$$W_k = \max_{\boldsymbol{\eta}\in\mathcal{C}_{\mathcal{K}}^{\mathcal{N}(k)}} \sum_{i\in\mathcal{K}} U'(R_i)\eta_i, \tag{2.5}$$

and the second is a user group scheduling problem

$$\max_{\sum t_k=1} \sum_k t_k W_k, \tag{2.6}$$

where W_k is the maximal weighted-sum-rate of user group k.

The solution to (2.6) is $t_{k^*} = 1$ if $W_{k^*} = \max_k W_k$. Thus, the key is to solve (2.5).

2.3.2 Weighted-Sum-Rate Maximization Problem

Note that if $i \notin \mathcal{N}(k)$, then $\eta_i = 0$. By slightly abusing the notation, denote $\boldsymbol{\eta}_k$ as the rate vector of users in user group $\mathcal{N}(k)$. By simplifying the notation, (2.5) is equivalent to

$$\max_{\boldsymbol{\eta} \in \mathcal{C}_{\mathcal{N}}^{\mathcal{N}}} \mathbf{u}^T \boldsymbol{\eta}. \tag{2.7}$$

By introducing an auxiliary variable power p and replacing the constraint $\boldsymbol{\eta} \in \mathcal{C}_{\mathcal{N}}^{\mathcal{N}}$ with $\boldsymbol{\eta} \in \mathcal{C}_{\mathcal{N}}^{\mathcal{N}}(p)$ and $p < P$, where $\mathcal{C}_{\mathcal{N}}^{\mathcal{N}}(p)$ is the achievable rate region with power constraint p of the N-user GBC, the partial dual problem of (2.7) is

$$\max_{\boldsymbol{\eta}, p} \mathbf{u}^T \boldsymbol{\eta} - \lambda p \quad s.t. \quad \boldsymbol{\eta} \in \mathcal{C}_{\mathcal{N}}^{\mathcal{N}}(p). \tag{2.8}$$

According to [82], the solution to problem (2.8) is

$$\eta_i^*(\mathbf{u}, \lambda) = \int_{\mathcal{A}_i} \frac{1}{n_i + z} dz,$$

where $\mathcal{A}_i = \{z \in [0, \infty) : u_i(z) = u^*(z)\}$, $u_i(z) = \frac{u_i}{n_i + z} - \lambda$, and $u^*(z) = [\max_i u_i(z)]^+$.

By writing the above solution explicitly, for problem (2.7) we have

$$\eta_i^* = \log \frac{n_i + U_i}{n_i + L_i}, \tag{2.9}$$

where

$$L_i = \begin{cases} \min\left(P, \max_{u_j < u_i} [\frac{u_i n_j - u_j n_i}{u_j - u_i}]^+\right), & u_i \neq \min_j u_j, \\ 0 & u_i = \min_j u_j, \end{cases}$$

and

$$U_i = \begin{cases} \min\left(P, \min_{u_j > u_i} [\frac{u_i n_j - u_j n_i}{u_j - u_i}]^+\right), & u_i \neq \max_j u_j, \\ P & u_i = \max_j u_j. \end{cases}$$

Note that, for any $i \neq j$, we need to find all the $\frac{u_i n_j - u_j n_i}{u_j - u_i}$, whose number is $N(N-1)/2$. Thus the computational complexity to solve the weighted-sum-rate maximization problem is $O(N(N-1)/2)$.

Algorithm 1 Opt Algorithm

1: **for all** $\mathcal{N}(k) \in \mathcal{S}$ **do**
2: solve

$$\mathbf{r}_k = \arg\max_{\boldsymbol{\eta} \in \mathcal{C}_{\mathcal{N}(k)}^{\mathcal{N}(k)}} \sum_{i \in \mathcal{N}(k)} U'(R_i)\eta_i$$

using (2.9).
3: obtain W_k based on (2.5).
4: **end for**
5: $k^* = \arg\max_k W_k$.
6: **Return:** $\mathcal{N}(k^*)$, $\mathbf{r}_{k^*}$.

2.4 Scheduling Algorithm

2.4.1 Optimal (Opt) Algorithm

The optimal scheduling algorithm is to find the user group $\mathcal{N}(k^*)$ with the maximal weighted-sum-rate W_{k^*}, according to (2.5) and (2.6). Thus we need to exhaustively search all the $|S|$ possible user groups, and calculate the corresponding W_k. The algorithm is shown in Algorithm 1.

When calculating W_k, $(u_i n_j - u_j n_i)/(u_j - u_i)$ is repeatedly calculated, so we can obtain a look-up table for $(u_i n_j - u_j n_i)/(u_j - u_i)$ to save the computation, which requires $K(K-1)/2$ calculations. Then, the complexity to solve (2.5) is linear w.r.t. N. Overall, the computational complexity is $O(|S|N + K(K-1)/2)$. Be aware that if N is small, constructing a look-up table is not efficient and costs more computation. While with moderator N, using a look-up table can save one order of magnitude computational complexity.

2.4.2 Iterative User Selection (IUS) ALgorithm

To reduce the computational complexity, an iterative user selection algorithm can be used. If $N = 1$, then only the user with maximal $U'(R_i)\eta_i$ will be selected, where $\eta_i = \log(1 + \gamma_i)$. Based on the selected first user 1^*, we search for the second user to maximize $U'(R_i)\eta_i + U'(R_{1^*})\eta_{1^*}$. Iteratively, we can find upto N users based on the previously selected users. Be aware that it is a greedy approach: selecting the user that can provide the maximal additional weighted rate gain. The algorithm is shown in Algorithm 2.

Algorithm 2 IUS Algorithm

1: Initialize $\mathcal{L}^{(0)} = \emptyset$ $\{\mathcal{L}^{(i)}$ is the selected user group after i-th iteration.$\}$
2: **for** $i = 1$ to N **do**
3: **for all** $k \in \mathcal{K} - \mathcal{L}^{(i-1)}$ **do**
4: $\mathcal{T}_k^{(i)} = \mathcal{L}^{(i-1)} \bigcup \{k\}$.
5: solve

$$W_k^{(i)} = \max \sum_{j \in \mathcal{T}_k^{(i)}} U'(R_j)\eta_j \quad \text{s.t.} \quad \boldsymbol{\eta} \in \mathcal{C}_{\mathcal{T}_k^{(i)}}^{\mathcal{T}_k^{(i)}}$$

using (2.9) and (2.5). The corresponding rate is $\mathbf{r}_k^{(i)}$.
6: **end for**
7: $k^* = \arg\max_k W_k^{(i)}$.
8: $\mathcal{L}^{(i)} = \mathcal{L}^{(i-1)} \bigcup \{k^*\}$
9: **end for**
10: **Return:** $\mathcal{L}^{(N)}$, $\mathbf{r}_{k^*}^{(N)}$.

In the i-th iteration, we need to search over $K - i + 1$ users to solve an i-user weighted-sum-rate maximization problem, and totally we have N iterations. By using the look-up table as in the Opt algorithm, the overall computational complexity is $O(\sum_{i=1}^{N} i(K - i + 1) + \frac{K(K-1)}{2}) = O(\frac{N(N+1)}{2}(K + 1 - \frac{2N+1}{3}) + \frac{K(K-1)}{2})$.

2.4.3 Random User Candidate Based Algorithm

The IUS algorithm can reduce the computational complexity, but cannot reduce the CSI feedback load, since the user is unaware whether it will be selected or not. To reduce the overhead, we randomly select N' users only to feedback their CSI. By replacing K with N' in the above obtained computational complexity, we can obtain the corresponding computational complexity of the random user candidate based algorithm. The performance of this naive approach can be considered as a lower-bound of the low-overhead algorithms.

2.4.4 Computational Complexity Comparison

The computational complexities of the proposed algorithms are compared in Fig.2.1. With the increment of N, the computational complexity of the Opt algorithm will first increase, then decrease. This is because when $N > K/2$, $|S|$ will decrease, *i.e.* the number of user group candidate will decrease. For the IUS algorithm, the computational complexity is increasing with the increment of N, and does not have

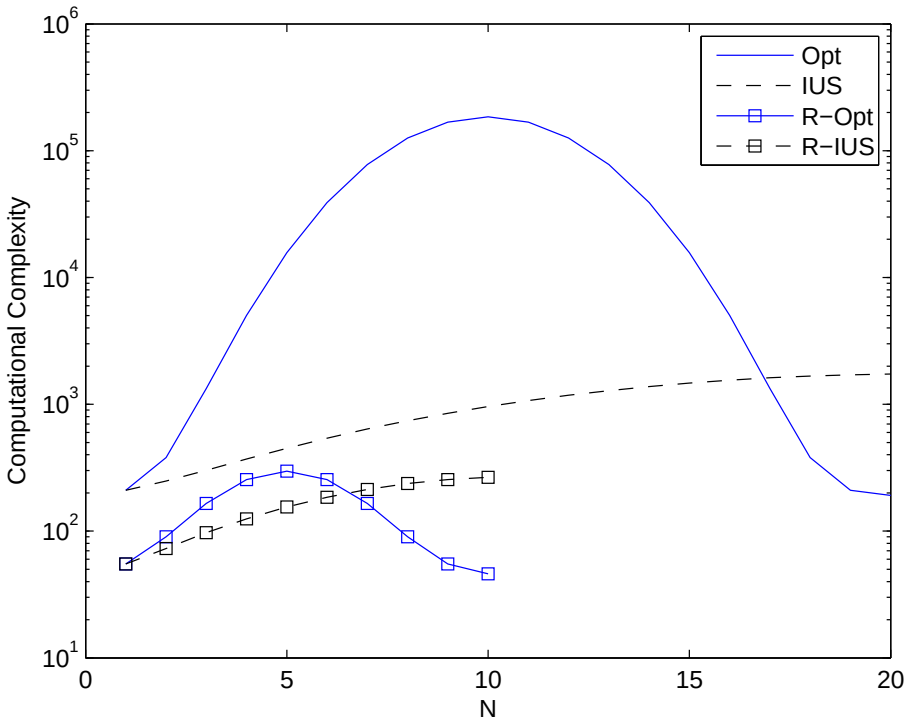


Figure 2.1: Comparison of Computational Complexity, $K = 20$, $N' = 10$.

the decreasing feature. When N is close to K, the computational complexity can be higher than the optimal solution. This is because the iterative user selection approach does not have the full knowledge of N, and when N is large, the IUS algorithm will test many unnecessary sub-user groups with the size less than N. For the random user candidate based algorithm, the computational complexity is scaled down significantly, compared with the corresponding original algorithm, especially for the Opt algorithm.

2.5 Performance Evaluation

In this section, extensive simulations are conducted to evaluate the performance of the proposed algorithms, several remarkable observations are presented.

2.5.1 Simulation Setting

Utility Function

The utility function chosen to be evaluated is the α-fairness [58],

$$U(x) = \begin{cases} \log(x), & \alpha = 1, \\ (1-\alpha)^{-1}x^{1-\alpha} & \text{others}, \end{cases}$$

where x is the average throughput, whose unit is bps/Hz in this chapter. The derivative is

$$U'(x) = x^{-\alpha}.$$

By choosing different α, the objective is to maximize the fairness measurement based on different principles, and the relative value of the measurement is of more interests. For instance, if $\alpha = 0$, then the objective is to maximize the system throughput; if $\alpha = 1$, then it is to maximize the proportional fairness; if $\alpha \to \infty$, then it is to maximize the max-min fairness.

Channel Model and Parameters

The raw SNR of user i can be modelled as the product of two random variables, *i.e.* $\gamma_i = a_i b_i$, where a_i represents the large-scale path loss and shadowing component, and b_i represents the small-scale fast-fading component. We assume that the envelop of the small-scale fast-fading component follows a Nakagami fading, so the distribution of b_i is a gamma distribution, *i.e.*

$$f(x) = (\frac{m}{P_r})^m \frac{x^{m-1}}{\Gamma(m)} \exp(-\frac{mx}{P_r}), \quad (m \geq 0.5)$$

where m is the fading parameter, P_r is the average received power in the Nakagami fading which is fixed to one. Note that, m is used to control the variability of b_i, and a small m results in a large variation of b_i. When $m = 1$, the Nakagami fading becomes the Rayleigh fading.

Other Parameters

Monte Carlo simulation is used to evaluate the system performance. The step size ϵ is used to control the accuracy and speed of convergence, and we use $\epsilon = 0.001$ in the simulation. The initial value of the estimated average throughput $\mathbf{R}$ also affects the convergence speed, and we choose it as $R_i = \log_2(1 + a_i)/K$. During the simulation, we find that roughly after 3000 fading blocks, $\mathbf{R}$ can weakly converge, while the average throughput obtained by calculating the average of $\mathbf{r}_{\mathcal{K}}$ converges much faster than $\mathbf{R}$. Thus, for each SNR, we run 10000 fading blocks, and collect the results from the last 5000 fading blocks.

2.5.2 Scenario One: Homogeneous Fading Channel

In a homogeneous AWGN broadcast channel (every user has the identical SNR), using SPC cannot result in an extra capacity region. We are interested in whether SPC can improve the system performance in a homogeneous broadcast block fading channel (the instantaneous SNR of each user is i.i.d.). Here we assume $\forall i, a_i = \gamma$. The system utility of a special case is compared in Table 2.1. The utilities obtained by Opt algorithm and IUS algorithm only have a small difference. The system utility almost does not change with the increment of N, and the utility difference is negligible. Also, such utility difference is irrelative to the SNR γ. This reflects that although in each fading block, using SPC can result in an extra capacity region, this instantaneous gain cannot result in a noticeable average gain. This also suggests under the peak power constraint in a homogeneous fading channel, using SPC cannot obviously improve the performance of a single-user point-to-point link, which is different from the results in [52], where the average power constraint is used.

2.5.3 Scenario Two: Heterogeneous Fading Channel

In this scenario, we assume that the large-scale path loss and shadowing component for different user is different, but it is fixed for each user. Specifically, we assume $a_i = a(i-1) + 1$ dB, where a is used to tune the SNR range of users.

Utility Comparison

First we compare the system utilities of the Opt algorithm and the IUS algorithm under different settings, and the results are shown in Table 2.2. With the increment

Table 2.1: System Utility Comparison: Homogeneous Broadcast Fading Channel. $m = 1$, $K = 10$, $\alpha = 1$

γ (dB)	5	10	15	20	25	30
	Optimal Algorithm					
$N = 1$	-11.24	-7.35	-4.43	-2.17	-0.28	1.27
$N = 2$	-11.24	-7.34	-4.43	-2.14	-0.27	1.30
$N = 3$	-11.22	-7.34	-4.43	-2.14	-0.27	1.29
$N = 4$	-11.21	-7.35	-4.43	-2.15	-0.28	1.30
$N = 5$	-11.21	-7.35	-4.43	-2.14	-0.29	1.29
	IUS					
$N = 1$	-11.25	-7.34	-4.44	-2.16	-0.28	1.28
$N = 2$	-11.22	-7.33	-4.42	-2.16	-0.27	1.31
$N = 3$	-11.22	-7.35	-4.42	-2.16	-0.27	1.30
$N = 4$	-11.24	-7.32	-4.41	-2.15	-0.26	1.30
$N = 5$	-11.23	-7.37	-4.44	-2.14	-0.29	1.30

of N, the system utility is increasing. When $a = 1$, $N = 2$ can provide almost optimal utility; while for $a = 3$, the utility is close to optimal when $N \geq 3$. This suggests that SPC can provide more gain when the SNR range of users is large, and in order to fully exploit such gain, the number of layers should also be large. Next, comparing different m, a large m means a small variability of the SNR, and results in a small utility. When m is larger, the utility gain provided by SPC is larger. This suggests that SPC is more valuable in a small-variation fast-fading channel. Considering the IUS algorithm, all the trends observed from the Opt algorithm are preserved, and its utility is slightly lower than that of the Opt algorithm.

The utilities of random user candidate based algorithms (R-Opt and R-IUS) are shown in Table 2.3. All the trends observed in Table 2.2 also exist, but the absolute value of utility is small when N' is small. In order to reduce the feedback overhead, the resulting utility loss is significant.

System Throughput Comparison

The system throughputs of the Opt algorithm and the IUS algorithm are compared with different utility functions, and the results are shown in Figs. 2.2 and 2.3. The

Table 2.2: System Utility Comparison: Heterogeneous Broadcast Fading Channel. $K = 10$, $\alpha = 1$

N			1	2	3	4	5
m=1	a=1	Opt	-11.01	-10.56	-10.55	-10.54	-10.54
		IUS	-11.01	-10.72	-10.73	-10.71	-10.71
	a=3	Opt	-5.66	-3.63	-3.35	-3.32	-3.32
		IUS	-5.66	-3.98	-3.96	-3.96	-3.97
m=10	a=3	Opt	-7.37	-4.86	-4.33	-4.19	-4.14
		IUS	-7.37	-5.09	-4.95	-4.96	-4.95

Table 2.3: System Utility of Random User Candidate Based Algorithm, $K = 10$, $\alpha = 1$, $m = 1$, $a = 3$.

N		1	2	3	4	5
$N' = 6$	R-Opt	-6.24	-4.43	-4.30	-4.25	-4.25
	R-IUS	-6.24	-4.85	-4.86	-4.83	-4.86
$N' = 8$	R-Opt	-5.88	-3.98	-3.76	-3.71	-3.69
	R-IUS	-5.89	-4.34	-4.34	-4.34	-4.35

unit of throughput is bps/Hz and is omitted in all figures.

In Fig. 2.2, the system throughput with $\alpha = 1$ is compared. The impact of fast fading is tuned by changing m, and the results show that SPC can provide a large gain for a large m. By comparing different a, the throughput of different SNR range is compared. When a is small, the throughput gain due to SPC is quite marginal. A larger N means more SPC layers and a higher complexity. In all cases, when $N > 2$, further increase N cannot provide significant throughput gain, which suggests $N = 2$ is a reasonable value.

Comparing Fig. 2.3 with Fig. 2.2, we can see all the trends observed when $\alpha = 1$ can also be observed when $\alpha = 10$, while the values are different since different objectives are used. By comparing the normalized SPC gain in the two figures, α has no great impact on it.

Comparing the results of the Opt algorithm and the IUS algorithm, whether the IUS algorithm has a larger throughput or not depends on α, while the throughput difference between the two algorithms is less than 2%.

Fig. 2.4 shows the system throughput of random user candidate based algorithms.

When $N' = 10$, the R-Opt and R-IUS become Opt and IUS respectively. By decreasing N', the performance loss is enlarged. Comparing different N, a larger N results in a larger performance loss, but the performance gains of using SPC with R-Opt/R-IUS are still substantial.

2.5.4 Summary

By investigating the performance under two scenarios, we have the following remarks. First, SPC cannot provide much gain in a homogeneous fading channel, when the system is subject to the peak power constraint. Second, the gain introduced by SPC depends on the variability of the channel. SPC can provide a higher gain when the channel experiences a less variable fast-fading. Third, how to determine the number of SPC layers mostly depends on the SNR range of users. Generally $N = 2$ or 3 is a reasonable value. Fourth, the normalized throughput gain provided by SPC is not highly related to the utility objective parameter α. Fifth, the performance of the IUS algorithm is close to that of the Opt algorithm, and the throughput difference is less than 2% in all simulated environments. When $N = 2$, their utility difference is negligible. Last, using random user candidate based algorithms will degrade the system performance, but they still can achieve a remarkable SPC gain.

2.6 Conclusion

In this chapter, we have investigated the resource allocation in a K-user wireless broadcast system with N-layer superposition coding. The problem has been formulated as a general sum-utility maximization problem where the utility is a function of the average throughput. Based on stochastic approximation and primal decomposition, the problem can be decomposed into two online problems: a user group selection problem and a weighted-sum-rate maximization problem. An optimal scheduling algorithm, a low complexity iterative user selection algorithm and a random user candidate based algorithm are proposed. Based on simulation, we find several important observations, which can be used as design guidelines for practical deployment of SPC (such as hierarchical modulation and network modulation [89, 8, 101]) in a multiuser wireless communication system.

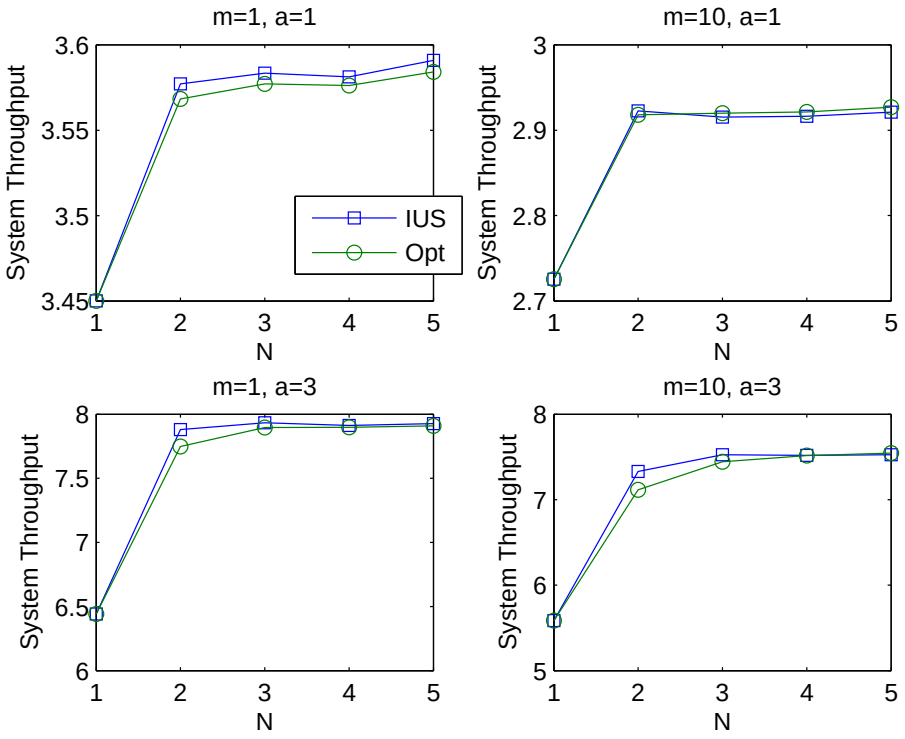


Figure 2.2: Comparison of System Throughput, $K = 10$, $\alpha = 1$.

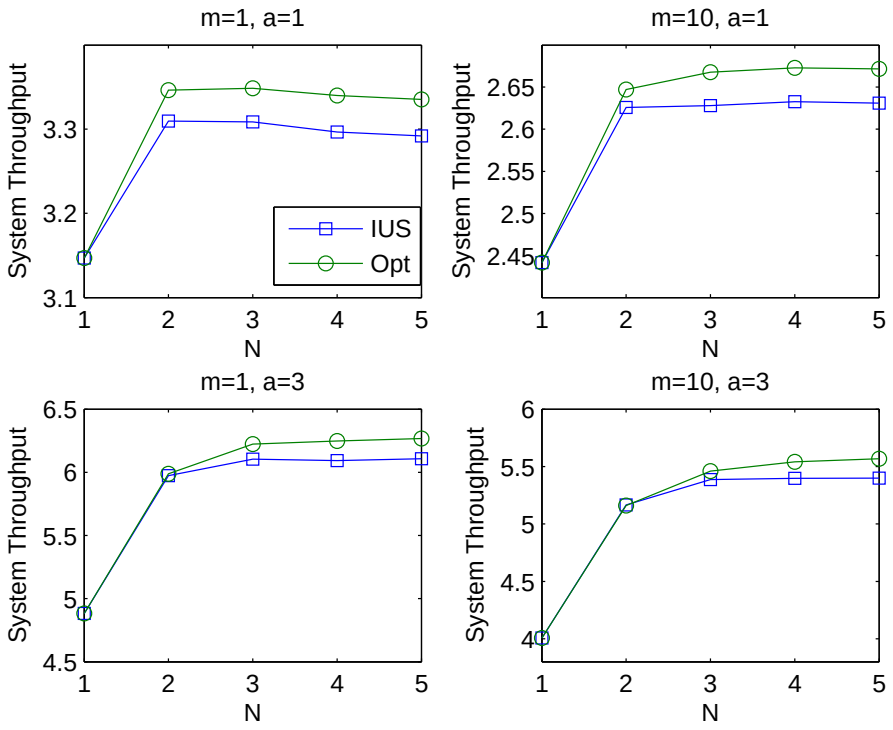


Figure 2.3: Comparison of System Throughput, $K = 10$, $\alpha = 10$.

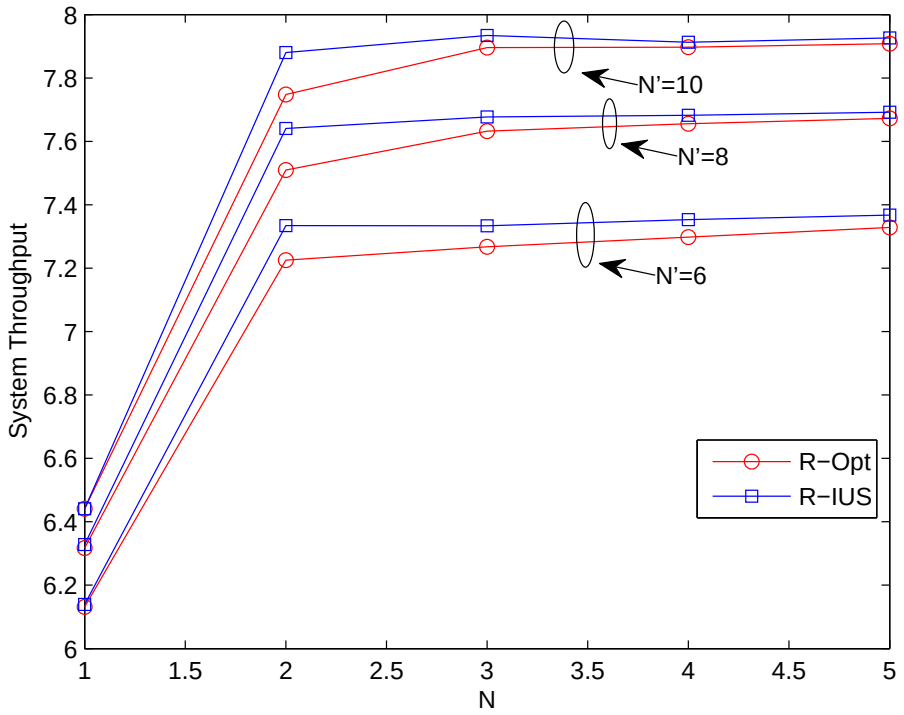


Figure 2.4: System Throughput of Random User Candidate Based Algorithm, $K = 10$, $\alpha = 1$, $m = 1$, $a = 3$.

Chapter 3

Proportional Fair Scheduling in Hierarchical Modulation Aided Wireless Networks

From Chapter 2, we know that SPC is more favourable in the scenario with small-variation fast-fading channel and a wide SNR range of users. In this chapter, we further study the SPC scheduling problem in a practical system. A practical implementation of SPC, hierarchical modulation (HM), has been adopted. We only consider the utility defined by proportional fairness as it is widely used in current wireless networks. An optimal algorithm and a low complexity suboptimal algorithm are proposed to solve the practical scheduling problem combining the opportunistic PFS and HM.

3.1 Introduction

Wireless channels are time-varying and broadcast in nature. How to optimize wireless scheduling algorithms to maximize system efficiency and ensure fairness, considering the wireless channel characteristics, is both challenging and promising.

Traditionally, a scheduler (in the link layer) divides the wireless resources into orthogonal logic links. For each logic link, the physical (PHY) layer deals with channel impairments (e.g., fading, shadowing, path loss) aimed to maximize the spectrum efficiency under certain bit error rate (BER) constraint. Using the services provided by the PHY layer, upper layer protocols can be designed without considering the

wireless channel characteristics.

Such layered solutions are not most efficient. In [39], an opportunistic scheduling was proposed for multiple users with independent, time-varying channel conditions sharing the uplink in a cellular network. The scheduler avoids to select the users in deep-fading to transmit to improve the system efficiency. Instead of concealing the fast-fading in the PHY layer, the opportunistic scheduling utilizes the randomness of channel conditions to improve system performance. This approach has also been extended for the downlink case in [83].

To ensure fairness among competing users, the proportionally fair (PF) rate allocation for wired networks was proposed in [35]. Using the channel state information (CSI) from the receiver, the opportunistic PF scheduling (PFS) algorithm for wireless networks was proposed in [85].

On the other hand, as discussed in chapter 2, SPC can achieve the capacity bound of a degraded Gaussian broadcast channel [10], and can improve the system performance in multi-user wireless systems. Recently, SPC has been implemented as HM using embedded constellation, which requires CSI from the receiver to select the suitable modulation.

Ideally, a wireless scheduler should exploit the multi-user diversity and the spatial diversity gain (using HM) while maintain fairness (using opportunistic PFS). Since the opportunistic PFS and HM both need the CSI feedback in a similar frequency in a block fading channel (per-block feedback), using these two technologies together, system performance can be improved by taking the advantage of both gains without increasing signaling message complexity. However, how to design an efficient and fair scheduler for HM-aided wireless networks is an open, challenging issue, since the user selection and resource allocation should be jointly optimized.

The main contributions of this chapter are three-fold. First, we formulate the two-user opportunistic PFS scheduling problems: an SPC-based theoretical problem using Shannon capacity and an HM-based practical problem. Second, we propose an optimal algorithm and a suboptimal algorithm to solve the practical scheduling problem using opportunistic PFS and HM. Third, extensive simulations have been conducted to evaluate the performance of the proposed algorithms. Simulation results have demonstrated that the proposed algorithms can achieve substantial throughput gain compared to the existing single-user PFS solution and have better fairness performance compared to the existing HM based solutions.

3.2 Preliminaries and Related Work

3.2.1 Proportional Fair Scheduling

In a wireless network, as the channels of different users are independent and heterogeneous, opportunistic scheduling was proposed to exploit the multi-user diversity gain to improve system efficiency [85, 99]. In a practical wireless system, a greedy algorithm that selects the user with the best channel quality to transmit tends to always select the users near the base station (BS) which is unfair and leads to the starvation problem.

The PFS was proposed to make the tradeoff between the multi-user diversity gain and fairness. A scheduling policy $\mathcal{P}$ is proportionally fair, if and only if the sum of the logarithmic average user throughput is maximized after the scheduling decision [85]:

$$\mathcal{P} = \arg\max_{\mathcal{S}} \sum_{i \in U} \log R_i^{(\mathcal{S})}, \tag{3.1}$$

where U is the active user set and $R_i^{(\mathcal{S})}$ is the average throughput of user i under scheduling policy $\mathcal{S}$.

If only one user is allowed to transmit in any time slot t, (3.1) is degenerated to selecting the user with the largest $r_i(t)/R_i(t)$ among all active users, where $r_i(t)$ is the instantaneous rate of user i in slot t, and $R_i(t)$ is its average throughput before slot t. Typically, the average throughput can be updated by an exponentially weighted moving average algorithm [85]:

$$R_i(t+1) = \begin{cases} (1-\frac{1}{T})R_i(t) + \frac{1}{T}r_i(t), & \text{for } i = i^*, \\ (1-\frac{1}{T})R_i(t), & \text{for } i \neq i^*, \end{cases} \tag{3.2}$$

where T corresponds to the window size to smooth the throughput, and i^* is the index of the scheduled user at time t. To simplify the notation, we omit (t) in $R(t)$ and $r(t)$ hereafter.

Similar to the results in [37] (which extended the PFS to a multi-carrier system), if in a system where multi-users can transmit simultaneously, the proportional fair scheduling policy $\mathcal{P}$ should satisfy

$$\mathcal{P} = \arg\max_{\mathcal{S}} \prod_{i \in U_S} (1 + \frac{r_i}{(T-1)R_i}), \tag{3.3}$$

where U_S is the set of the selected users by scheduling policy $\mathcal{S}$.

3.2.2 Superposition Coding

Superposing signals for multiple users to achieve the capacity of degraded Gaussian broadcast channel was first introduced in [10] and named SPC. SPC is of great interest to enhance the downlink performance in various scenarios [74, 92]. By using SPC together with SIC, the capacity bound of a downlink degraded Gaussian broadcast channel can be achieved [19]. In a general order $|h_1| \leq |h_2| \leq ... \leq |h_N|$, where $|h_i|$ is the channel gain of user i, the capacity bound of user i is defined by

$$r_i = \log_2 (1 + \frac{P_i |h_i|^2}{N_0 B + \sum_{j=i+1}^{K} P_j |h_i|^2}) \text{ bps/Hz}, \tag{3.4}$$

where P_i is the power allocated to user i, N_0 is the noise spectral density and B is the channel bandwidth.

3.2.3 Related Work

Although SPC and PFS have been proposed in 1970's and 1990's, respectively, there is very limited cross-domain work on the combination of these two powerful techniques. Recently, an implementation of SPC, called HM, has been adopted in Digital Video Broadcasting (DVB) and several other standards [79], which generates new interests in this promising area.

Different from the SPC-related problem [61], the scheduling problem for HM is more difficult due to the discrete feature of the number of bits allocated and their BER requirements. In [25, 26], two opportunistic scheduling algorithms were proposed for a wireless network using two-layer HM, which allows two users to transmit simultaneously, namely two-best-user opportunistic scheduling (TBS) and hybrid two-user opportunistic scheduling (HTS) respectively. TBS selects two users with the first and second highest channel gain to transmit; HTS selects the first user with the highest channel gain and the second user with the highest relative channel gain, defined as the instantaneous channel gain normalized to its short-term average channel gain. TBS can be viewed as a direct extension of the single-user throughput-maximized opportunistic scheduling, and HTS is aimed to achieve a better max-min fairness. However, the bit allocation scheme in TBS/HTS may not fully explore the benefit of SPC/HM, since the constellation size is determined by the first user only.

In [33], a hybrid two-user opportunistic scheduling algorithm named HTS2 was proposed. It selects the first user based on the proportional fairness and the second user based on whether there are higher channel gain users in the same subcarrier. After user selection, transmission power is reallocated among the selected users to achieve better fairness.

The above HM-based scheduling algorithms separated the user selection and power allocation, which cannot achieve proportional fairness. In [1], an analytical approach was given to study PFS with SPC, jointly considering user selection and power allocation according to the sum-rate gain. Since multi-user PFS is not necessarily lead to a sum-rate gain, the resultant user selection may deviate from the optimal one based on multi-user PFS.

3.3 System Model and Problem Formulation

3.3.1 System Model

We consider a two-layer SPC-aided single-cell wireless cellular network. The channel is assumed to be a quasi-static flat fading channel, i.e., in each time slot, the channels are static and independent of each other, and among different time slots, the channel fading follows a specific probability distribution (such as Rayleigh fading). We focus on the downlink case, and assume that the BS can obtain the instantaneous CSI for each slot.

3.3.2 Proportional Fair Scheduling Problem

With HM, a scheduler can allocate a slot to at most two receivers. According to (3.3), the PFS problem is to find the user pair (i^*, j^*) that satisfies $(i^*, j^*) = \arg\max_{(i,j)} U_{(i,j)}$ at each time slot, where $U_{(i,j)}$ is the PF-utility of the selected user pair (i, j).

Based on (3.3), we define $U_{(i,j)}$ as

$$U_{(i,j)} = \begin{cases} (1 + \frac{r^i_{(i,j)}}{(T-1)R_i})(1 + \frac{r^j_{(i,j)}}{(T-1)R_j}), & \text{if } i \neq j, \\ 1 + \frac{r^i_{(i,j)}}{(T-1)R_i}, & \text{if } i = j, \end{cases} \tag{3.5}$$

where $r^i_{(i,j)}$ is the instantaneous rate of user i when user pair (i, j) is selected. For $i \neq j$, w.l.o.g, user i is assumed to have a higher or equal channel gain than user j.

3.3.3 Theoretical Capacity Based PF-Utility Maximization Problem

First, we consider the theoretical Shannon capacity based PF-utility maximization problem when using PFS in SPC-aided wireless networks.

The PF-utility maximization problem is to maximize the PF-utility function (3.5) where the instantaneous rates lie in the capacity region. Since the capacity region is a closed convex set and the PF-utility function (3.5) is also convex, the problem is to maximize a convex function on a closed convex set. The maximum can be obtained in the boundary of the convex set [66]. Thus, for our problem, full power should be allocated to maximize the PF-utility.

Define the channel SNR of user i as $\gamma_i = P|h_i|^2/N_0B$, where P is the system power constraint. Based on (3.4), the instantaneous rates of paired user i and j can be written as, respectively,

$$r^i_{(i,j)} = \log(1 + q^i_{(i,j)}\gamma_i), \tag{3.6}$$

$$r^j_{(i,j)} = \log(1 + \gamma_j) - \log(1 + q^i_{(i,j)}\gamma_j), \tag{3.7}$$

where $q^i_{(i,j)} = P_i/P$ is the portion of power allocated to user i.

For $i \neq j$, by substituting (3.6) and (3.7) into (3.5), after some manipulations and simplifications, the PF-utility maximization problem is formulated as follows.

Problem 3.1.

$$\begin{aligned} \max \quad & \frac{\log(1 + q^i_{(i,j)}\gamma_i)}{R_i} - \frac{\log(1 + q^i_{(i,j)}\gamma_j)}{R_j} \\ & + \frac{\log(1 + q^i_{(i,j)}\gamma_i)\log(\frac{1+\gamma_j}{1+q^i_{(i,j)}\gamma_j})}{(T-1)R_iR_j}, \\ \text{s.t.} \quad & 0 \leq q^i_{(i,j)} \leq 1. \end{aligned} \tag{3.8}$$

When $q^i_{(i,j)} = 0$ or 1 the Problem 3.1 also includes the case that a single user is scheduled.

Due to the duality of Gaussian multiple-access and broadcast channels [34], Problem 3.1 also formulates the two-user SIC based multi-user uplink scheduling problem when the sum uplink power is fixed. In practice, individual power constraint is a more realistic assumption, and our approach is not directly applicable and needs further extension.

3.3.4 HM-Based PF-Utility Maximization Problem

The optimal solution of Problem 3.1 may not be feasible in a practical system, and an uncoded HM based system is considered in this subsection. Since in an uncoded system with block fading, BER and BLER (Block Error Rate) have a direct one-to-one mapping, in the following, we only consider the PF-Utility maximization in the symbol level.

The HM-based PAM (or QAM) is a generalized PAM (or QAM) with flexible Euclidean distance among constellation points. Here, we consider a system deploying two-layer HM based square QAM (HMsQAM) with Gray mapping, which has been well investigated in [86] and the reference therein. Since square QAM can be viewed as two identical and orthogonal PAM modulations, we first analyze the two-layer HM based PAM (HMPAM) with Gray mapping, having n bits in the first layer and $m-n$ bits in the second layer, named $2^n/2^m$-HMPAM.

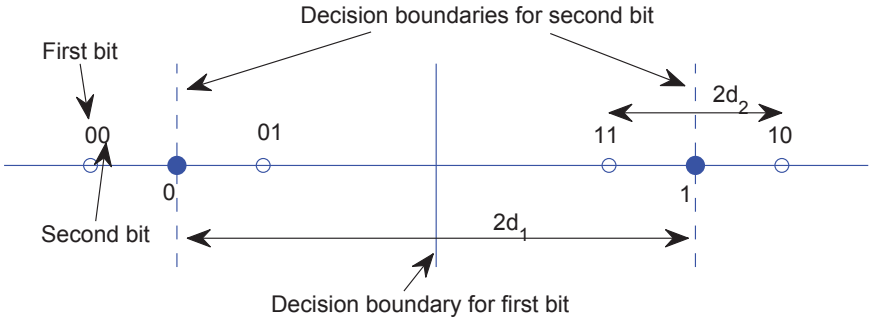


Figure 3.1: 2/4-HMPAM with Gray mapping. The filled circles represent the fictitious symbols which are not actually transmitted. The open circles represent the real transmitted symbols. The digits attached to the symbols represent the bits information of the symbols (real or fictitious).

As discussed in [86], a $2^n/2^m$-HMPAM has m levels of constellation points. The constellation points in level-i ($i < m$) are fictitious and represent the symbols corresponding to the i-th bit. The constellation points in level-m represent real symbols. The Euclidean distance between the constellation points in level-i is $2d_i$. The first n bits belongs to the first layer and the rest $m-n$ bits belong to the second layer. Within each layer, we have $d_i = 2d_{i+1}$ (here level-i and level-$(i+1)$ belong to the same layer.). A sample of 2/4-HMPAM is shown in Fig. 3.1, which has two layers and also two levels.

The Euclidean distance between the constellation points in level-i of $2^n/2^m$-HMPAM

is

$$2d_i = \begin{cases} (\frac{1}{2})^{i-1} 2d_1, & \text{for } i \leq n, \\ (\frac{1}{2})^{i-n-1} 2d_{n+1}, & \text{for } n+1 \leq i \leq m. \end{cases} \tag{3.9}$$

The exact closed-form BER expression of generalized PAM and QAM is derived in [86]. Based on the constellation diagram of each modulation scheme, the decision region can be obtained for each bit. Thus by taking the integral over the decision region for each bit, the probability that the bit is decoded successfully as well as the error probability can be obtained. As the exact BER expression is very complicated, it is not easy to be used to formulate and solve our problem, and also the computational complexity of the resultant algorithm will be too high for an online scheduler. So, a BER approximation is needed. Since the exact BER expression is a summation of a series of complementary error function erfc(), which decades very fast, thus the BER is mainly determined by the shortest Euclidean distance between the corresponding constellation points to the decision boundaries.

For instance, the BER of the first and second bit in a 2/4-HMPAM are, respectively

$$P_{b,2}^{(1)} = \frac{1}{4}[\text{erfc}\,(\frac{d_1+d_2}{\sqrt{N_0}}) + \text{erfc}\,(\frac{d_1-d_2}{\sqrt{N_0}})],$$

$$P_{b,2}^{(2)} = \frac{1}{2}\,\text{erfc}\,(\frac{d_2}{\sqrt{N_0}}) + \frac{1}{4}[\text{erfc}\,(\frac{2d_1-d_2}{\sqrt{N_0}}) - \text{erfc}\,(\frac{2d_1+d_2}{\sqrt{N_0}})].$$

The BER of the first bit is mainly determined by the Euclidean distance between symbol 01 and 11, and similarly, the BER of the second bit is by that between symbol 00 and 01 (or symbol 11 and 10). Hence, their BER can be approximated by, respectively,

$$\tilde{P}_{b,4}^{(1)} = \frac{1}{4}\,\text{erfc}\,(\frac{d_1-d_2}{\sqrt{N_0}}), \tilde{P}_{b,4}^{(2)} = \frac{1}{2}\,\text{erfc}\,(\frac{d_2}{\sqrt{N_0}}).$$

Fig. 3.2 shows the BER approximation for 2/4-PAM. From the figure, in all three configurations of 2/4-PAM, the approximations are close to the analytical results.

Such approximation only considers the dominant term in the exact BER expression, which can be named dominant term approximation. Given that the typical BER

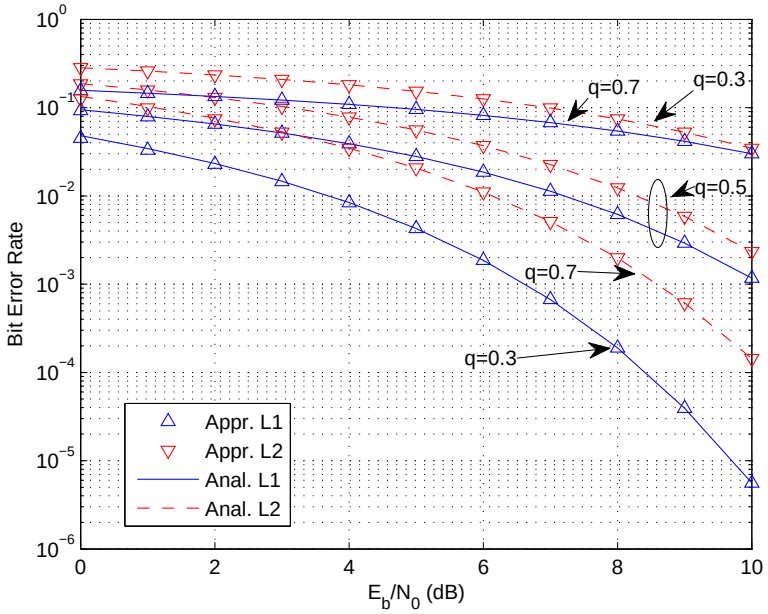


Figure 3.2: Validation of BER approximation for 2/4-HMPAM with Gray mapping. q is the energy portion of layer-1 signal. By different q, we have different constellation diagram setting, which has different Euclidean distance among constellation points. The choice of q is limited since $\forall i < j, d_i > d_j$.

requirement is below 10^{-3}, it is reasonable to use the dominant term approximation, whose accuracy has been confirmed in [100] for QAM in the low BER region.

Using the dominant term approximation and the constellation point distance (3.9), the BER of each bit in $2^{2n}/2^{2m}$-HMsQAM can be calculated by

$$\tilde{P}_{b,m}^{(i)} = \begin{cases} \frac{1}{2^{m+1-i}} \operatorname{erfc}\left(\frac{d_i - \sum_{j=i+1}^{m} d_j}{\sqrt{N_0}}\right), & \text{for } i = 1, ..., m-1, \\ \frac{1}{2} \operatorname{erfc}\left(\frac{d_i}{\sqrt{N_0}}\right), & \text{for } i = m, \end{cases} \tag{3.10}$$

where $\tilde{P}_{b,m}^{(i)}$ is the BER of the i-th bit in in-phase or quadrature and m is the total number of bits transmitted in in-phase or quadrature. As layer-1 bits and layer-2 bits are transmitted to different users, whose received signal powers can be different, *i.e.*, the Euclidean distances of the received signals of different users are different, thus we need to differentiate the corresponding Euclidean distances. In the following, d_i and $d_{i,(j)}$ ($j \in \{1, 2\}$) are used to represent the Euclidean distance in the transmitted signal constellation diagram and that in the received signal constellation diagram of layer-j user, respectively.

The BER of the layer-1 bits is

$$\tilde{P}_e^{(1)} = \frac{1}{n}\sum_{i=1}^{n} \tilde{P}_{b,m}^{(i)}. \tag{3.11}$$

Substituting (3.10) into (3.11) with $d_{i,(1)}$ replacing d_i, after some manipulations, we obtain

$$\tilde{P}_e^{(1)} = \frac{1}{n}[(\frac{1}{2})^{m-n} - (\frac{1}{2})^m] \operatorname{erfc}(\frac{d_{n,(1)} - (2^{m-n}-1)d_{m,(1)}}{\sqrt{N_0}}). \tag{3.12}$$

Similarly, the BER of the layer-2 bits is

$$\tilde{P}_e^{(2)} = \frac{1}{m-n}[1 - (\frac{1}{2})^{m-n}] \operatorname{erfc}(\frac{d_{m,(2)}}{\sqrt{N_0}}). \tag{3.13}$$

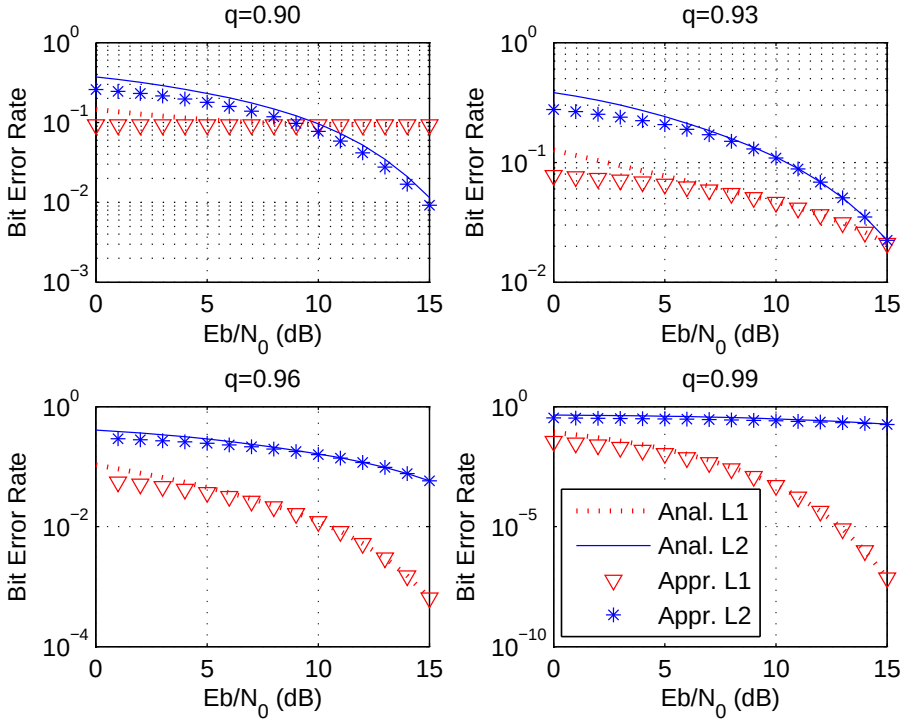


Figure 3.3: Validation of BER approximation for 4/16-HMPAM with Gray mapping.

A sample validation of BER based on the dominant term approximation for 4/16-PAM is shown in Fig. 3.3. The dominant term approximation is accurate as long as the BER is lower than a threshold, *e.g.*, 10^{-3}.

The average symbol energy of the transmitted signal is $E_s^{(t)} = 2\sum_{i=1}^{m} d_i^2$, which consists two parts, the average symbol energy of layer-1 signal and that of layer-2 signal,

$$E_{s,1}^{(t)} = 2\sum_{i=1}^{n} d_i^2, \tag{3.14}$$

$$E_{s,2}^{(t)} = 2\sum_{i=n+1}^{m} d_i^2. \tag{3.15}$$

For the average symbol energy of the transmitted signal and the received signal, we have

$$\frac{E_{s,j}^{(t)} B N_0}{P} = \frac{E_{s,j}^{(r)}}{\gamma^{(j)}}, \quad j = 1, 2, \tag{3.16}$$

where $\gamma^{(j)}$ is the channel SNR of layer-j user.

Since the transmitted signal should satisfy the energy constraint, we have $E_s^{(t)} \leq P/B$. By substituting (3.9), (3.14), (3.15) and (3.16) into the energy constraint, we obtain

$$\frac{4^{m-n}-1}{3}(\frac{d_{m,(2)}}{\sqrt{N_0}})^2\frac{2}{\gamma^{(2)}} + \frac{4^n-1}{3}(\frac{d_{n,(1)}}{\sqrt{N_0}})^2\frac{2}{\gamma^{(1)}} \leq 1. \tag{3.17}$$

Substituting (3.9) and (3.16) into (3.11), the BER of layer-1 bits is

$$\tilde{P}_e^{(1)} = \frac{2^n-1}{n2^m}\,\mathrm{erfc}\,(\frac{d_{n,(1)}}{\sqrt{N_0}} - (2^{m-n}-1)\sqrt{\frac{\gamma^{(1)}}{\gamma^{(2)}}}\frac{d_{m,(2)}}{\sqrt{N_0}}). \tag{3.18}$$

Thus, the PF-utility maximization problem for a system with HMsQAM modulation can be formulated as follows.

Problem 3.2.

$$\begin{aligned} \max \quad & (1+\frac{2n}{(T-1)R^{(1)}})(1+\frac{2(m-n)}{(T-1)R^{(2)}}) \\ \text{s.t.} \quad & \tilde{P}_e^{(1)} < P_{e1}, \tilde{P}_e^{(2)} < P_{e2} \quad \textit{and} \quad (3.17), \\ & m \geq n, \quad m, n \in [0, 1, ..., K], \end{aligned}$$

where $R^{(j)}$ is the average rate of layer-j user, P_{e1} and P_{e2} are the BER requirements

of user 1 and 2, respectively, and 2^{2K}-HMsQAM is the maximal modulation scheme.

Using the dominant term approximation, the BER of the higher layer (generally for users with higher channel SNR) does not rely on the constellation distance in the lower layer, which dramatically simplifies the problem, not only for BER calculation, but also for modulation configuration.

3.4 Scheduling Algorithm Design

For the optimal scheduling algorithm design, the convexity of Problem 1 may not be equivalent to the convexity of PF-utility function (3.5) and is generally unknown. Since Problem 1 is a one-dimensional problem in a closed set, the optimum solution can be obtained straightforwardly by comparing the points with zero derivative and the boundary points. In the following, we mainly focus on Problem 3.2, which is a more practical and difficult problem.

3.4.1 Optimal Solution: O^{2U} HM PFS Algorithm

Problem 3.2 is a mixed-integer programming problem which is generally hard to solve. However as the number of the available modulation schemes is limited in practice, it is feasible to solve such a problem by the searching algorithm shown in Algorithm 3. Based on possible bit allocation to different layer and the BER requirement of layer-2, we can obtain the Euclidean distance requirement of layer-2 signal (lines 2 - 8). Based on the energy constraint and the BER constraint of layer-1 signal (lines 9 -11), we can decide whether it is a feasible solution. If it is feasible, then the utility is calculated and updated (lines 12 - 14). By searching all the possible bit allocation, we can find one with the maximal utility, which is the desired one.

Using Algorithm 3 to maximize the PF-utility of any given user pair, the scheduling problem can be solved by Algorithm 4, which is a simple exhaustive search over all user pairs. Note that in Algorithm 4, we do not explicitly calculate the maximal PF-utility when a single user is chosen to transmit. This is because the single user PF-utility can be calculated with Algorithm 3 when $n = 0$ or $m = n$.

Both Algorithm 3 and Algorithm 4 can be further improved in reducing the computation time. For instance, if a BER look-up table is used to deal with the single user case in Algorithm 3, the computation time can be slightly reduced, but the order of computational complexity remains the same. Thus, we only present the original

Algorithm 3 MaxUtil

Require: $(R^{(1)}, R^{(2)}, P_{e1}, P_{e2}, \gamma^{(1)}, \gamma^{(2)}, T)$ $\{R^{(1)}$ and $R^{(2)}$ are the average throughput of user 1 and user 2 respectively. P_{e1} and P_{e2} are the required BERs of user 1 and user 2 respectively. $\gamma^{(1)}$ and $\gamma^{(2)}$ are the channel SNRs of user 1 and user 2 respectively. T is the window size to average the throughput.$\}$

$U = 0$
for $n = 0$ to K **do**
  for $m = n$ to K **do**
    if $m - n = 0$ **then**
      $\frac{d_{m,(2)}}{\sqrt{N_0}} = 0$
    else
      calculate $\frac{d_{m,(2)}}{\sqrt{N_0}}$ based on $\tilde{P}_e^{(2)} < P_{e2}$;
    end if
    if (3.17) is feasible to solve **then**
      calculate $\frac{d_{m,(1)}}{\sqrt{N_0}}$;
      if $\frac{d_{n,(1)}}{\sqrt{N_0}} - (2^{m-n} - 1)\sqrt{\frac{\gamma^{(1)}}{\gamma^{(2)}}}\frac{d_{m,(2)}}{\sqrt{N_0}} > 0$ **and** $\tilde{P}_e^{(1)} < P_{e1}$ **or** $n = 0$ **then**
        if $U(n,m) = (1 + \frac{2n}{(T-1)R^{(1)}})(1 + \frac{2(m-n)}{(T-1)R^{(2)}}) > U$ **then**
          $U = U(n,m)$
          set (n,m) as the index of HMsQAM modulation scheme
        end if
      end if
    end if
  end for
end for
Return: U, (n,m)

algorithm here.

The computational complexity of Algorithm 3 is $\mathcal{O}(K^2)$, where K is the number of available modulation schemes, and it is generally a small number (typically less than 10 in a practical system). The computational complexity of Algorithm 4 is $\mathcal{O}(N^2K^2)$, where N is the number of active users and can be a large number, so lower complexity algorithms with comparable performance to Algorithm 4 are desirable for the online scheduler.

3.4.2 Suboptimal Solution: S^{2U} HM PFS Algorithm

In order to reduce the computational complexity of the proposed optimal algorithm, we reconsider the PF-utility maximization Problem 3.1. Note that if T is sufficiently large, the problem is approximated by a maximal weighted sum rate problem as follows.

Algorithm 4 Optimal Two-User PFS with SPC

Require: $(\{R_i : i \in S\}, \{\gamma_i : i \in S\}, P_{e1}, P_{e2}, T)$ $\{R_i$ is the average throughput of user i. S is the set of user. P_e is the required BER. γ_i is the channel SNR of user i. T is the window size to average the throughput.$\}$

$U = 0$, $(n, m) = (0, 0), I = (0, 0)$
sort users according to their channel SNR. Using index (i) to represent the user with i-th lowest channel SNR.
for $i = 1$ to N **do**
  for $j = i + 1$ to N **do**
    $(U', (n', m')) = \text{MaxUtil}(R_{(i)}, R_{(j)}, P_e, P_e, \gamma_{(i)}, \gamma_{(j)}, T)$
    if $U' > U$ **then**
      $(n, m) = (n', m')$, $U = U'$
      if $n = 0$ **then**
        set $I = ((j), (j))$ as the user index.
      else if $m = n$ **then**
        set $I = ((i), (i))$ as the user index.
      else
        set $I = ((i), (j))$ as the user index.
      end if
    end if
  end for
end for
Return: I, (n, m)

Problem 3.3.

$$\begin{aligned} \max \quad & \frac{\log(1 + q^i_{(i,j)}\gamma_i)}{R_i} - \frac{\log(1 + q^i_{(i,j)}\gamma_j)}{R_j} \\ \text{s.t.} \quad & 0 \le q^i_{(i,j)} \le 1 \end{aligned}$$

By taking the derivative of the objective function and setting it to zero, we can obtain the optimal power allocation $q^{i*}_{(i,j)}$ and the corresponding maximal PF-utility $U^*_{(i,j)}$. When two users are selected, i.e., $q^{i*}_{(i,j)} \in (0, 1)$, the maximal PF-utility is

$$U^*_{(i,j)} = \frac{\log(1 + \gamma_j)}{R_j} + G^{\text{SPC}}_{i,j}, \tag{3.19}$$

where

$$G^{\text{SPC}}_{i,j} = \frac{1}{R_i}\log(\frac{R_j}{\gamma_j}\frac{\gamma_i - \gamma_j}{R_i - R_j}) - \frac{1}{R_j}\log(\frac{R_i}{\gamma_i}\frac{\gamma_i - \gamma_j}{R_i - R_j})$$

is the SPC gain.

Algorithm 5 Suboptimal Two-User PFS with SPC

Require: ($\{R_i : i \in S\}$, $\{\gamma_i : i \in S\}$, P_{e1}, P_{e2}, T) {R_i is the average user rates of user i. S is the set of user. P_e is the required BER. γ_i is the channel SNR of user i. T is the window size to average the throughput.}

1: $U = 0$, $(n, m) = (0, 0), I = (0, 0)$
2: sort users according to their channel SNR. Using index (i) to represent the user with i-th lowest channel SNR.
3: find $i^* = \arg\max_i \log(1 + \gamma_i)/R_i$
4: find $(k) = i^*$
5: **if** $k = N$ **then**
6: find n based on BER Lookup Table. $I = (i^*, i^*)$.
7: **end if**
8: **for** $j = k + 1$ to N **do**
9: $(U', (n', m')) = \text{MaxUtil}(R_{(i^*)}, R_{(j)}, P_e, P_e, \gamma_{(i^*)}, \gamma_{(j)}, T)$
10: **if** $U' > U$ **then**
11: $(n, m) = (n', m')$, $U = U'$
12: **if** $n = 0$ **then**
13: set $I = ((j), (j))$ as the user index.
14: **else if** $m = n$ **then**
15: set $I = ((i^*), (i^*))$ as the user index.
16: **else**
17: set $I = ((i^*), (j))$ as the user index.
18: **end if**
19: **end if**
20: **end for**
21: **Return:** I, (n, m)

Recall that the single user PFS is to select the user with the largest $\log(1+\gamma)/R$, which is identical to the first term of (3.19). Thus, we can separate the user selection and resource allocation by using single-user PFS to find the first user, which is the low SNR user (lines 2 - 4) and by using multi-user PFS to find the second user which is the high SNR user (lines 8 - 20). Note that such decomposition cannot guarantee the optimality, since the largest $\log(1+\gamma)/R$ may not necessarily lead to the maximal PF-utility among all possible user pairs, so it is a suboptimal solution. Following a similar approach, a heuristic suboptimal algorithm can be developed to solve Problem 3.2, as shown in Algorithm 5. Note that, if the user selected by single-user PFS has the maximal channel SNR, only one user will be selected and the bit allocation can be done by a single-user BER look-up table (lines 5 - 7). The computational complexity of Algorithm 5 is $\mathcal{O}(NK^2)$ only, which is one order lower than that of Algorithm 4.

3.4.3 Further Discussion: J-Layer HM problem

We can further consider a more general scheduling problem with J-layer HM, which becomes much more complicated as discussed below. To find the optimal solution, we need to search all the possible resource allocation strategies and test whether it is feasible under the energy and BER constraints. By observing the procedure of BER approximation, we can conclude that, the expression of the approximated BER of a higher layer is not related to the lower layers. Thus, for any bit allocation scheme (allocating the number of bits to each layer (user)), we can tell whether the bit allocation is feasible by iteratively solving the BER constraints and test the energy constraint. The computational complexity for a given group of users is at least $\Omega(\log(K)^J)$ for the J-Layer HM. To find the optimal utility, we need to search all $\frac{N!}{J!(N-J)!}$ user groups, thus the overall computational complexity is at least $\Omega(\log(K)^J \frac{N!}{J!(N-J)!})$. Note that, without the BER approximation, the problem is even much harder since it is not easy to test whether the bit allocation is feasible due to the coupled BER expressions for different layers.

We can also generalize the sub-optimal Algorithm 5 for J-layer HM, by selecting the layer-1 user based on the single-user PFS first, then selecting the higher layer's users sequentially based on the previously selected users and the i-user PFS criteria. On average, the number of candidate users for the first layer user is N, and the number of candidate users for the second layer user is $N/2$ (since only the users having higher SNRs than the first user will be searched), and the number of candidate users for the J-th layer user is $N/2^{J-1}$. Thus, on average the number of candidate user groups is $\frac{N^J}{2^{(J-1)J/2}}$. The average computational complexity is then at least $\Omega(\log(K)^J \frac{N^J}{2^{(J-1)J/2}})$, which is much smaller compared to that of the optimal solution. Obviously, this heuristic solution is suboptimal and its performance requires further investigation.

3.5 Performance Evaluation

Extensive simulations have been conducted to evaluate the performance of the proposed algorithms. For comparison, the performance of four existing state-of-the-art scheduling algorithms, including the single-user PFS [85], HTS [26], TBS [25] and HTS2 [1] is also evaluated by simulations.

Table 3.1: Parameter Setting.

Parameters	C	β	$\sigma_{\phi_{dB}}$	l_0	h_t
Values	-31.45dB	3.71	3.5dB	1m	10m
Parameters	h_r	P_t	N_0	B	r
Values	1m	10dBm	-174 dBm/Hz	500 KHz	300m

3.5.1 Simulation Setting

We consider the downlink of a single-cell narrowband cellular network and assume the wireless channel is a quasi-static Rayleigh fading channel. A classic path loss and shadowing model is used to model the large-scale propagation effects [19]: $\frac{P_r}{P_t}(dB) = 10\log_{10} C - 10\beta \log_{10} \frac{l}{l_0} - \phi_{dB}$, where P_r is the received power, P_t is the transmitted power, C is a unit-less constant which depends on the characteristics of the antenna and the average channel attenuation, β is the path loss exponent, l is the distance between the transmitter and the receiver, l_0 is the reference distance, and ϕ_{dB} is a Gaussian distributed random variable with zero mean and variance of $\sigma^2_{\phi_{dB}}$.

Table 3.1 summarizes the parameters of an urban macro-cell given in [19] and other parameters used in simulation, including antenna height of the transmitter (h_t) and the receiver (h_r), maximal transmission power (P_t), noise spectral density (N_0), channel bandwidth (B) and radius of cell (r). The single-user modulation schemes considered are 2^{2n}-QAM and 2^n-PAM, $\forall n \in \{1, 2, 3, 4\}$, and the multi-user modulation schemes are $2^{2n}/2^{2m}$-HMsQAM, $\forall n, m \in \{1, 2, 3, 4\}, m > n$. The BER requirements are all set to be 10^{-3}. The simulation evaluates 400 random user deployments and each deployment has 2000 time slots. The throughput for each deployment is calculated by taking the average of 2000 time slots. The window size T for the PFS algorithms is 1000 as suggested in [32]. We also vary the number of users, N_u, to simulate a wide range of scenarios.

3.5.2 System PF-Utility Comparison

In Sec. 3.4, we propose an optimal algorithm and a suboptimal algorithm to solve Problem 3.2. The optimality of the algorithms can be judged by the system PF-utility. From the definition, the maximal system PF-utility leads to the best proportional fairness, which measures the tradeoff between efficiency and fairness. The system PF-utilities are compared in Fig. 3.4, where the y-axis is the system PF-utility and the

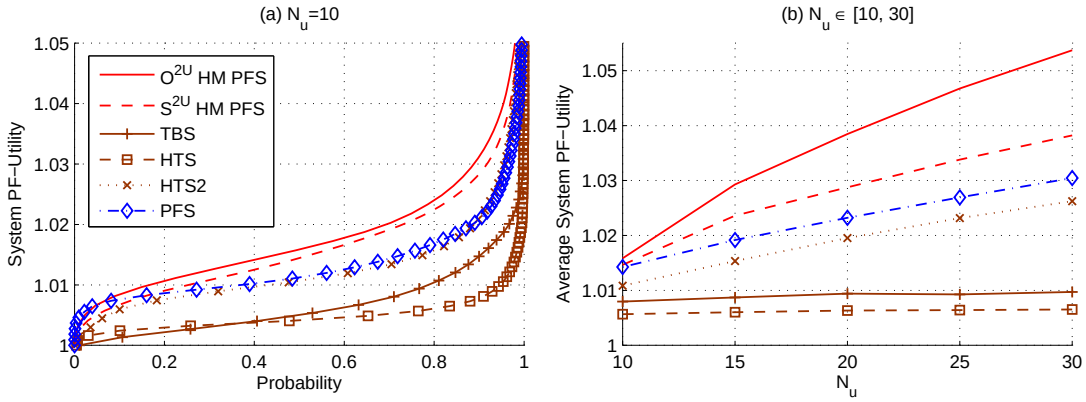


Figure 3.4: System PF-utility Comparison.

x-axis represents the probability that the PF-utility is below certain value. A higher curve indicates a higher PF-utility. As shown in Fig. 3.4 (a), the system PF-utility of the optimal O^{2U} HM PFS is strictly higher than that of the suboptimal S^{2U} HM PFS. Both of the proposed algorithms outperform the single-user PFS substantially, as they can use HM to explore the extra capacity, which moves the system operating point towards a higher PF-utility. On the other hand, the system PF-utilities of the proposed scheduling algorithms are much higher than that of HTS and TBS, the two greedy scheduling algorithms for HM transmissions. Thus, without PFS, even though HTS and TBS can use HM to achieve a higher throughput for some users, the overall system PF-utility is not improved. The proposed algorithms also outperform HTS2 which is the best multi-user scheduling algorithm in the literature. Although HTS2 can utilize the extra capacity region thanks to HM, it is unable to achieve the best PF-utility, due to the choice of the second user. The average system PF-utilities w.r.t. the number of users are compared in Fig. 3.4 (b). It can be seen that, the PF-utilities of PFS-based algorithms are higher with the increment of N_u. For the two greedy algorithms, TBS and HTS, N_u has almost no positive impact on the PF-utility. Next, we compare their performance in terms of fairness and system throughput separately.

3.5.3 Fairness Comparison

Jain's Fairness Index

Jain's fairness index has been widely used to measure the fairness, which is represented as $\mathcal{J}(x_1, x_2, \ldots, x_{N_u}) = \frac{(\sum_{i=1}^{N_u} x_i)^2}{N_u \cdot \sum_{i=1}^{N_u} x_i^2}$, where x_i is user i's throughput.

In Fig. 3.5 (a), where the y-axis is the Jain's fairness index in terms of per-

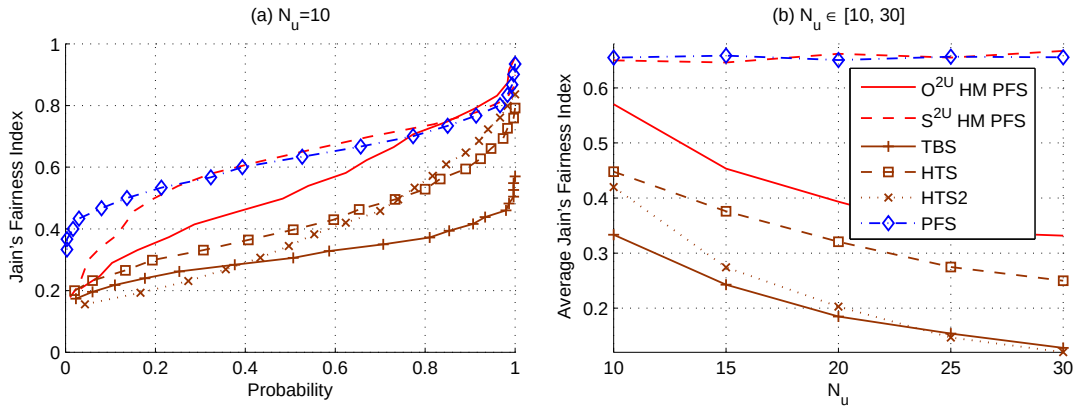


Figure 3.5: Jain's Fairness Index Comparison.

user throughput and the x-axis is its Cumulative Distribution Function (CDF). Each point in the curve illustrates the probability that the fairness index is below certain threshold. The scheduler corresponding to a higher curve achieves better fairness. Overall, the proposed HM PFS and PFS schemes have the best fairness performance among all multi-user scheduling algorithms. For the TBS, HTS, and HTS2 algorithms, they all tend to favor the users with better channel quality, which results in worse fairness performance.

The average Jain's fairness index w.r.t. the number of users is compared in Fig. 3.5 (b). The Jain's index of all schemes except PFS and O^{2U} HM PFS will decrease with a larger number of users, which means more unfairness for the users with a low SNR. The PFS and O^{2U} HM PFS can maintain a good Jain's fairness independent of the number of users.

The limitation of Jain's fairness index is that it does not consider the heterogeneity of users, and it is not desirable for the system where the user is not charged solely based on the throughput. In a practical system, the distribution of per-user throughput is a more meaningful metric for both fairness and user-perceived QoS. Next, we consider the CDF function of per-user throughput.

CDF Function of Per-User Throughput

If the performance of the worst-user is improved, there will be more satisfied users with higher overall user-perceived QoS. Given the performance of the $\alpha\%$ worst users (who have the throughput lower than $1-\alpha\%$ of the total users), we can judge whether the scheduler is worst-user friendly or not.

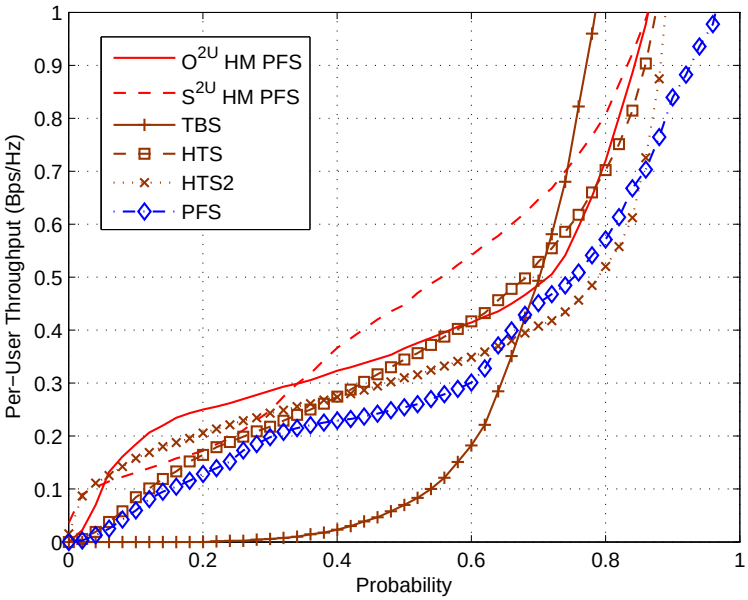


Figure 3.6: Per-user Throughput Distribution, $N_u = 10$.

In Fig. 3.6, the y-axis represents the per-user throughput, and the x-axis is its CDF. As shown in the figure, with the TBS algorithm, about 30% users in the system can be starving (with zero throughput), as TBS focuses on maximizing the total throughput only. Compared to TBS, HTS uses a different second-user selection criterion which significantly reduces the percentage of starving users, so its performance is close to PFS. HTS2, which is a modified algorithm of HTS, can achieve an even better performance than that of PFS when α is small. However, the per-user throughput of about 80% of users under HTS2 is below 0.5 bps/Hz, which is worse than PFS.

Overall, in terms of α% worst-user friendliness, the proposed O^{2U} HM PFS and S^{2U} HM PFS algorithms are the best two algorithms, for majority of cases when $\alpha \leq 60$. It means that the users with worse channel conditions can achieve higher throughput using the proposed two algorithms than the previous solutions. The question is whether they achieve the worst-user friendliness at the cost of the system efficiency, which will be investigated in the following subsection.

3.5.4 System Throughput

Here, system throughput is used to measure the efficiency. The comparison is based on two network configurations with different number of users N_u, and the results are

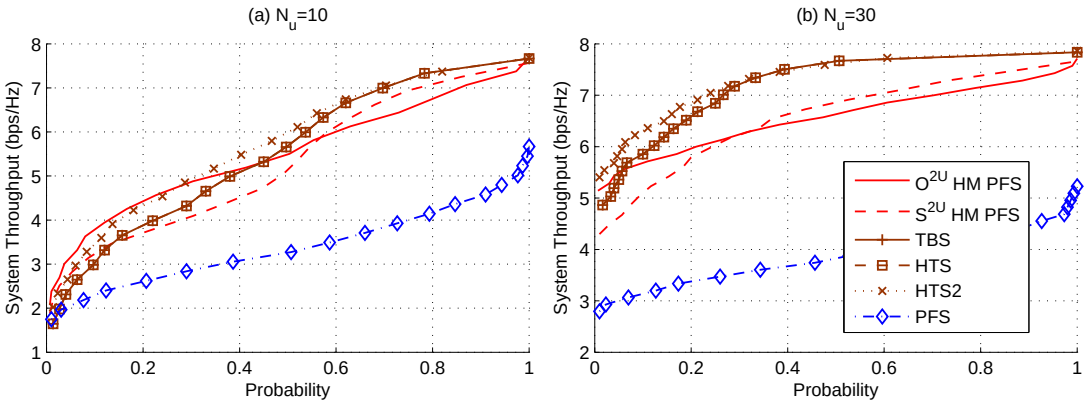


Figure 3.7: System Throughput.

shown in Fig. 3.7. The y-axis represents the system throughput, and the x-axis is its CDF, and a higher curve corresponding to better throughput performance. In Fig. 3.7 (a), the two best-user selection based greedy scheduling algorithms, HTS and TBS, achieve almost identical throughput so the two curves overlap with each other. Since both of the schemes cannot fully explore the benefit of SPC/HM, the throughput is not always the best. The HTS2 achieves the highest throughput for about 60% of the cases. The throughputs of TBS, HTS and HTS2 converge in the tail part. O^{2U} HM PFS can achieve the highest system throughput for 30% cases. The average system throughput gap between S^{2U} HM PFS and O^{2U} HM PFS is less than 5%.

Comparing the results in Figs. 3.7 (a) and (b), with the increment of the number of users, N_u, using the four multiuser scheduling algorithms can substantially increase the system throughput (more than 50% on average). The throughput increment of TBS, HTS and HTS2 are larger than the two proposed PFS algorithms. This is because a greedy algorithm tends to select the user with the best channel quality, which leads to a higher throughput. Comparing Figs. 3.7 (a) and 3.6, the higher system throughput of HTS2 is contributed mainly by 15% of the best users, which concludes the fairness concern for HTS2.

In summary, by using multi-user scheduling with HM, the system throughput can be improved substantially. For instance, as shown in Fig. 3.7 (b), on average, the system throughput with HM is increased by 80% compared with the case without HM (i.e., single-user PFS). Furthermore, the proposed O^{2U} HM PFS and S^{2U} HM PFS can achieve a comparable system throughput as the other three multi-user scheduling algorithms, TBS, HTS and HTS2. Different from TBS, HTS and HTS2, which suffer

from starvation and unfairness problems, the proposed scheduling algorithms can maintain good fairness, so users can experience more consistent and satisfactory QoS.

3.5.5 Access Delay

Beside the throughput and fairness, delay is also an important QoS metric. The link-layer delay includes queueing delay, transmission delay and access delay. The queueing delay and transmission delay depend on the per-user throughput, i.e., a higher throughput means a higher service rate, which results in a smaller queueing plus transmission delay. Since the proposed algorithms are worst-user friendly in terms of per-user throughput, they should be worst-user friendly in terms of queueing and transmission delay too. Thus, here we focus on the access delay.

Access delay is defined as the number of time slots that a head-of-queue packet has to wait till it is transmitted. A large access delay may have adverse impacts on upper-layer protocol performance, such as leading to a time-out event of a TCP connection.

The average access delay performance is compared in Fig. 3.8. The y-axis represents the average access delay of a user, and the x-axis is its CDF. Overall, the access delay using a multi-user scheduling (except TBS) can be reduced as two users are scheduled to transmit per-slot. TBS suffers severe unfairness and starvation problem: around 30% of the users have a very small access delay while the majority of the users have the access delay even longer than that with single-user PFS.

3.6 Conclusion

In this chapter, we have investigated the opportunistic PFS scheduling in an HM-aided wireless network. A Shannon capacity-based theoretical utility maximization problem considering SPC and an HM-based utility maximization problem have been formulated. The former provides insights on theoretical system performance bounds and guidelines for solving the latter which is the focus of this work. An optimal scheduling algorithm O^{2U} HM PFS and a low-complexity suboptimal scheduling algorithm S^{2U} HM PFS have been proposed. We have evaluated the performance of the proposed algorithms, compared with other state-of-the-art HM based multi-user scheduling algorithms and the single-user opportunistic PFS algorithm. Simulation results have shown that both O^{2U} HM PFS and S^{2U} HM PFS can increase the system

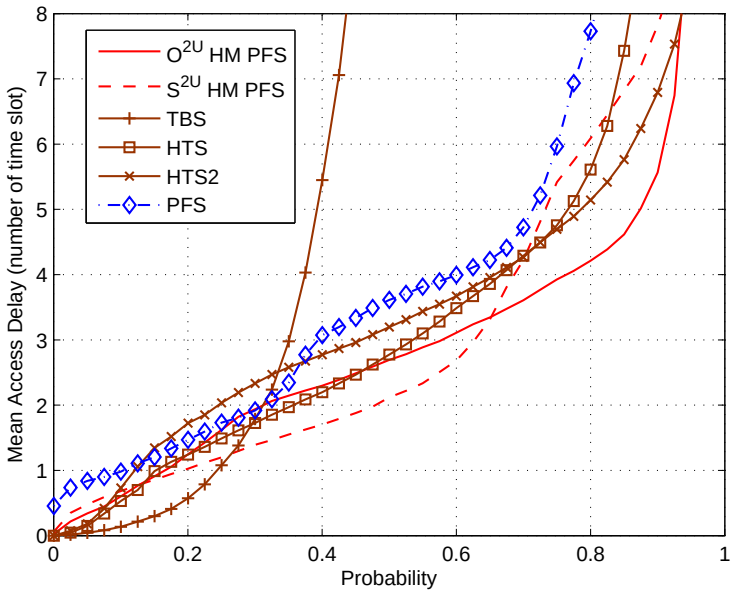


Figure 3.8: Access Delay, $N_u = 10$.

throughput substantially, while the fairness which is critical to user-perceived QoS can be well maintained.

Chapter 4

Stability Region of Opportunistic Scheduling in Wireless Networks

In previous chapters, we have discussed special utility-based scheduling algorithms in SPC/HM-aided saturated wireless systems. In this chapter, we study the behavior of utility-based scheduling algorithms in a general unsaturated wireless systems with stochastic traffic. It is pointed in the literature that the utility-based scheduling may not be able to stabilize the system under some circumstances even if the arrival rate lies inside the capacity region. Thus it is important to quantify the stability region of these utility-based scheduling algorithms proposed based on saturated system assumption. By treating an intermediate variable differently, the utility-based scheduling previously proposed can be classified into two types: the utility-based (UB) scheduling and the channel-rate-based (CRB) scheduling. The UB scheduling is a generalized proportional fair scheduling in an unsaturated system, and the CRB scheduling is a variant of the UB scheduling. We give the closed-form expression of the stability region of the CRB scheduling, and a numerical method to obtain the stability region of the UB scheduling. Both of the two scheduling policies are not throughput-optimal, and thus in general their stability regions are smaller than the ergodic capacity region. With the CRB scheduling, the stability region is a convex hull, while with the UB scheduling, the stability region is generally even non-convex and may exhibit some undesirable properties, such as decreasing the traffic of one flow may lead another flow being unstable, and proportionally decreasing the traffic of all flows may lead a stable system to be unstable. We further show that, as long as the arrival rate lies inside the ergodic capacity region, we can assign a proper weight

to each user, and based on the weighted UB/CRB scheduling policies, the system can be stabilized. Detailed numerical examples and simulations are given to illustrate the stability region of the two policies and validate our analysis.

4.1 Introduction and Related Work

In [96], the utility-based scheduling was proposed as a generalization of the opportunistic scheduling proposed in [39, 83], whose objective is to maximize a pre-defined utility based on the long-term achievable throughput. Based on the stochastic approximation, the convergence of such policy is guaranteed under a mild condition [77, 43]. The work has been further extended to different network scenarios, such as cooperative networks [38], or networks with different wireless techniques, such as the downlink and the uplink of an orthogonal frequency-division multiplexing (OFDM) system [95, 27, 28].

All these works designed the scheduler based on an assumption that each user always has sufficient data to transmit. The assumption simplifies the problem, but as shown in [2], these kinds of schedulers may lead the system to be unstable, while the system in the same circumstance can be stabilized by other scheduling policies, such as max-weight scheduling [80]. The key reason here is because, without considering the stochastic characteristic of incoming traffic, although the arrival rate lies inside the ergodic capacity region, the tie-breaking rule used in the above utility-based scheduling policies is not efficient as these policies schedule some users too frequently and lose the chance to explore the multi-user diversity gain, and thus they are not throughput-optimal.

Little work has been done in quantifying the stability region of the opportunistic scheduling policies. The stability region of an opportunistic scheduling policy in a two-user wireless network with i.i.d. Bernoulli arrival traffic was derived in [23]. Different from the general utility-based scheduling, the scheduler discussed in [23] is a normalized SNR one, where the user is scheduled based on the normalized instantaneous SNR. The author observed that the stability region is less than the ergodic capacity region, while by varying the normalized factor, the union of the resultant stability region is equal to the ergodic capacity region. Note that, with the identical normalized factor, the scheduler is able to explore the maximal multi-user diversity, but is not easy to explore other features, such as fairness. By changing the normalized factor, the fairness feature can be implicitly explored, while it is unclear how to design

the normalized factor for a specific fairness objective. Also, the prior knowledge assumption of the channel in [23] may bring difficulty to implement such policy. In [67], the authors discussed the two-user stability region in a static channel configuration with concurrent transmissions. The scheduler discussed is a partial distributed scheduler, combining the user coordination with an Aloha media access control (MAC), which may not be a suitable choice for a centralized wireless network due to the low channel efficiency of the Aloha MAC.

In this chapter, we quantify the stability region of two opportunistic scheduling policies with a general traffic arrival in a wireless system with N users. The two scheduling policies include a UB one and a CRB one. The CRB scheduling can be viewed as a variant of the UB scheduling, by treating an intermediate control variable differently. For the UB scheduling, the explicit closed-form stability region generally cannot be obtained, while we develop a theorem to examine the stability of a system given the arrival rate, and a numerical method is provided to obtain the stability region in a two-user system. We further study the properties of the stability region of the UB scheduling, and show that it is generally non-convex and may also exhibit some undesirable features. For instance, decreasing the arrival rate of one user may lead the system to be unstable. For the CRB scheduling, we obtain the closed-form expression of the stability region, which is a convex hull. Besides the stability region, we further study the extended stability region by giving a weight to each user. The results show that by varying the weight assigned to each user, the union of the resultant stability region is equal to the ergodic capacity region, for both scheduling policies. This suggests as long as the system can be stabilized, by assigning a proper weight to each user, using a non-throughput-optimal scheduling may also stabilize the system.

It is further noted that, the results of the CRB scheduling is similar to the work in [23], while our work is more general. We use a more general traffic model, consider a general N user system, and discuss a scheduling algorithm that can be easily designed to achieve certain utility objective.

In the following of this chapter, bold face letters represent vectors, and calligraphic letters represent sets.

4.2 System Models

4.2.1 Channel Model

The system has one server who has packets to transmit through a shared wireless channel to N independent users. The set of users is denoted by $\mathcal{N} = \{1, 2, ..., N\}$. The power set of $\mathcal{N}$ is denoted by $\mathbb{S}$, and the cardinality of $\mathbb{S}$ is $|\mathbb{S}| = 2^N$. We use $\mathcal{S}_i$ to denote the i-th element in $\mathbb{S}$.

We assume that the shared wireless channel is time slotted block fading channel. The set of channel state is finite, which is represented as $\mathcal{M} = \{1, 2, ..., M\}$. Within each time slot, the channel state is constant. Crossing time slots, certain rule is used to govern the transition of the channel state. There is a vector of rates $\mathbf{u}^m = (u_1^m, u_2^m, ..., u_N^m)$ associated with each channel state $m \in \mathcal{M}$. The element $u_i^m \in \mathbb{N} \bigcup \{0\}$ means the number of packets that can be transmitted if the time slot is all allocated to user i in state m.

We further assume that the shared wireless channel state process is an irreducible discrete-time Markov chain with the state space $\mathcal{M}$. The stationary distribution of this Markov chain is denoted as $\boldsymbol{\pi} = (\pi^1, \pi^2, ..., \pi^M)$.

The capacity region of the system in state m is denoted as

$$\mathcal{C}_{\mathcal{N}}^m = \bigcup_{\sum_i t_i^m = 1} (u_1^m t_1^m, ..., u_N^m t_N^m),$$

where t_i^m is the time portion allocated to user i in state m.

The ergodic capacity region of the system is obtained as:

$$\bar{\mathcal{C}}_{\mathcal{N}} = \bigcup_{\sum_i t_i^m = 1} (\sum_m u_1^m t_1^m \pi^m, ..., \sum_m u_N^m t_N^m \pi^m). \tag{4.1}$$

Given the user set $\mathcal{A}$, the corresponding capacity region in state m and the ergodic capacity region can be obtained by assigning $t_i^m = 0$ for all $i \notin \mathcal{A}$, and are denoted by $\mathcal{C}_{\mathcal{A}}^m$ and $\bar{\mathcal{C}}_{\mathcal{A}}$ respectively. Denote $C_{\mathcal{A}}(t)$ as the capacity region of user set $\mathcal{A}$ in time slot t. Since state m and time slot t are associated, so if the state in t is m, we have $\mathcal{C}_{\mathcal{A}}^m = C_{\mathcal{A}}(t)$.

4.2.2 Queueing Model

Data packets are arrived randomly and queued up in an infinite buffer reserved for each user. The packet arrival process is considered as a stationary ergodic stochastic process with finite moments. The state of the i-th buffer is the queue length and denoted by $q_i(t)$. All queue states form a vector $\mathbf{q}(t) \in \mathbb{R}_+^N$, and are updated by

$$\mathbf{q}(t+1) = [\mathbf{q}(t) - \mathbf{r}(t) + \mathbf{a}(t)]^+, \tag{4.2}$$

where $[x]_i^+ = \max\{0, x_i\}$,$\forall i \in \mathcal{N}$, $\mathbf{r}(t) \in \mathbb{R}_+^N$ is the amount of transmitted data that is determined by the scheduling decision, and $\mathbf{a}(t) \in \mathbb{R}_+^N$ is the amount of arrived data in time t, which is a bounded random variable. The average arrival and service rates are $\boldsymbol{\lambda} = \mathbb{E}_t[\mathbf{a}(t)]$ and $\boldsymbol{\mu} = \mathbb{E}_t[\mathbf{r}(t)]$, respectively.

4.2.3 Scheduling Policy

We assume that at the beginning of each time slot, the server can observe the state of the channel and allocate the resource based on the observation.

Under the assumption that each user always has enough data to transmit, a utility-based scheduling policy, which is a generalized proportional fair scheduling [96, 43], allocates the rate to user in time slot t based on the following problem:

$$\mathbf{r}(t) = \arg\max_{\boldsymbol{\eta} \in \mathcal{C}_{\mathcal{N}}(t)} \sum_{i \in \mathcal{N}} f(R_i(t))\eta_i, \tag{4.3}$$

with ties being broken randomly, where function f is a derivative of a strictly concave smooth utility function U, $R_i(t)$ is the smoothed rate measurement of user i in time slot t, which can be updated by an exponentially weighted moving average algorithm [43]

$$\mathbf{R}(t) = \mathbf{R}(t-1) + \epsilon(\mathbf{r}(t) - \mathbf{R}(t-1)),$$

where ϵ is the step size.

According to [77, 43], by choosing a proper step size ϵ, $\mathbf{R}(t)$ weakly converges to the average allocated rate $\mathbf{R}_{\mathcal{N}}$ which can be obtained based on the following problem

$$\mathbf{R}_{\mathcal{N}} = \arg\max_{\boldsymbol{\eta} \in \bar{\mathcal{C}}_{\mathcal{N}}} \sum_{i \in \mathcal{N}} U(\eta_i).$$

Note that the online algorithm (4.3) cannot be directly used in a system without the assumption of enough backlogs, since it may allocate the resource to users with no packet to transmit. With some modifications to (4.3), two scheduling policies, the UB and the CRB scheduling, can be obtained for a system with stochastic arrival traffic.

The UB Scheduling

In time slot t, a user set $\mathcal{A}(t)$ is selected satisfying the condition that the queue length of each user in $\mathcal{A}(t)$ is sufficiently large, for instance $q_i(t) \geq q_i^{th}$, where q_i^{th} is the queue length threshold for user i. This treatment avoids the wireless resource been wasted that choosing a user without enough data to transmit. The specific value of q_i^{th} does not affect the stability region, as long as it is sufficiently large. With such treatment, the queue length dynamic in (4.2) becomes

$$\mathbf{q}(t+1) = \mathbf{q}(t) - \mathbf{r}(t) + \mathbf{a}(t). \tag{4.4}$$

Then the rate allocated to the user in $\mathcal{A}(t)$ is

$$\mathbf{r}_{\mathcal{A}(t)}^{\mathrm{UB}}(t) = \arg\max_{\boldsymbol{\eta} \in \mathcal{C}_{\mathcal{A}(t)}(t)} \sum_{i \in \mathcal{A}(t)} f(R_i^{\mathrm{UB}}(t))\eta_i, \tag{4.5}$$

with ties being broken randomly, and the rate allocated to the user in $\mathcal{N}|\mathcal{A}(t)$ is 0. Using $\mathbf{r}^{\mathrm{UB}}(t)$ to denote the allocated rate in time slot t, then R_i^{UB} is updated based on

$$\mathbf{R}^{\mathrm{UB}}(t) = \mathbf{R}^{\mathrm{UB}}(t-1) + \epsilon(\mathbf{r}^{\mathrm{UB}}(t) - \mathbf{R}^{\mathrm{UB}}(t-1)),$$

which is used to track the average throughput of the system.

The CRB Scheduling

For the CRB scheduling, in time slot t, based on the same method as the UB scheduling, we select the candidate user set $\mathcal{A}(t)$. The rate allocated to the user in $\mathcal{A}(t)$ is based on

$$\mathbf{r}_{\mathcal{A}(t)}^{\mathrm{CRB}}(t) = \arg\max_{\boldsymbol{\eta} \in \mathcal{C}_{\mathcal{A}(t)}(t)} \sum_{i \in \mathcal{A}(t)} f(R_i^{\mathrm{CRB}}(t))\eta_i, \tag{4.6}$$

with ties being broken randomly, and the rate allocated to user in $\mathcal{N}|\mathcal{A}(t)$ is 0. We use $\mathbf{r}^{\mathrm{CRB}}(t)$ to denote the allocated rate in time slot t.

Different from the UB scheduling, in the CRB scheduling, $\mathbf{R}^{\mathrm{CRB}}(t)$ is used to track the average channel-rate, and is updated by

$$\mathbf{R}^{\mathrm{CRB}}(t) = \mathbf{R}^{\mathrm{CRB}}(t-1) + \epsilon(\mathbf{r}(t) - \mathbf{R}^{\mathrm{CRB}}(t-1)),$$

where $\mathbf{r}(t)$ is the solution to (4.3).

How to update $\mathbf{R}^{\mathrm{UB}}(t)$ and $\mathbf{R}^{\mathrm{CRB}}(t)$ is the only difference between the UB and the CRB scheduling policies. For the CRB scheduling, the update is independent of the scheduling decision, while for the UB scheduling, the update depends on the scheduling decision in each time slot.

As shown in [43], under a mild condition, $\mathbf{R}^{\mathrm{UB}}(t)$ and $\mathbf{R}^{\mathrm{CRB}}(t)$ are both weakly converge. In the following, we only consider the case that $\mathbf{R}^{\mathrm{UB}}(t)$ and $\mathbf{R}^{\mathrm{CRB}}(t)$ converge.

By abusing the notation a bit, we also use $\mathbf{R}_{\mathcal{A}}$ to denote the rate vector of N users and satisfies $\forall j \notin \mathcal{A}$, $R_j = 0$, *i.e.*, $\mathbf{R}_{\mathcal{A}}^T = [\mathbf{R}_{\mathcal{A}}^T \quad \mathbf{R}_{\mathcal{N}|\mathcal{A}}^T]$, where $\mathbf{R}_{\mathcal{N}|\mathcal{A}} = \mathbf{0}$.

4.2.4 Stability

We apply the stability definition as it is used in [47].

Definition 4.1. A system of queues is said to be strongly stable if

$$\lim_{t\to\infty} \sup \mathbb{E}[\|\mathbf{q}(t)\|] < \infty,$$

where $\|\mathbf{q}(t)\|$ is the norm of vector $\mathbf{q}(t)$.

Since we only consider the case that $\mathbf{R}^{\mathrm{CRB}}(t)$ or $\mathbf{R}^{\mathrm{UB}}(t)$ converges, and after the convergence of $\mathbf{R}^{\mathrm{CRB}}(t)$ or $\mathbf{R}^{\mathrm{UB}}(t)$, the scheduling decision is only related to the current channel state and the queue state. Therefore, we can simplify the stability condition.

First, for the CRB scheduling we assume that at time slot 0, $\mathbf{R}^{\mathrm{CRB}}(t)$ has converged. Due to (4.4), when t is sufficiently large, we have

$$\begin{aligned} \mathbf{q}(t) &= \mathbf{q}(0) - \sum_{\tau=0}^{t-1} \mathbf{r}(\tau) + \sum_{\tau=0}^{t-1} \mathbf{a}(\tau) \\ &= \mathbf{q}(0) - \boldsymbol{\mu} t + \boldsymbol{\lambda} t \geq 0, \end{aligned}$$

which suggests for all i, $\lambda_i \geq \mu_i$.

Since the dimension of $\mathbf{q}(t)$ is finite, here we only consider L_1 norm of $\mathbf{q}(t)$, and we have

$$\|\mathbf{q}(t)\| = t\sum_i(\lambda_i - \mu_i) + \sum_i q_i(0).$$

Therefore

$$\lim_{t\to\infty}\sup \mathbb{E}[\|\mathbf{q}(t)\|] < \infty,$$

requires

$$\begin{aligned} & \lim_{t\to\infty}\sup \mathbb{E}[t\sum_i(\lambda_i - \mu_i) + \sum_i q_i(0)] \\ = & \lim_{t\to\infty} t\sum_i(\lambda_i - \mu_i) + \sum_i \mathbb{E}[q_i(0)] \leq \infty, \end{aligned}$$

which suggests $\sum_i(\lambda_i - \mu_i) \leq 0$.

In summary, the stability of the system requires $\boldsymbol{\lambda} = \boldsymbol{\mu}$, *i.e.*, the average arrival rate is identical to the average throughput. For the UB scheduling, based on the same argument, we can have the same result.

We further define the stability region of the system as follows:

Definition 4.2. The stability region of a system with scheduling policy p is defined as Λ^p, and we have $\forall\boldsymbol{\lambda} \in \Lambda^p$, the system is strongly stable; $\forall\boldsymbol{\lambda} \notin \Lambda^p$, the system is not strongly stable;

Without confusing, we also use the stability region of scheduling policy p to refer to the stability region of a system with scheduling policy p.

4.3 Stability Region of the CRB scheduling

We first tackle a simple case, the static channel case ($M = 1$), to obtain the stability region. Thereafter, the general stochastic channel case is discussed. We show that by replacing the capacity region with the ergodic capacity region, all the discussions for the static channel case also hold for the stochastic channel case.

4.3.1 Static Channel Case

Since the channel only has one state, we have $\forall \mathcal{A} \in \mathbb{S}$, $\bar{\mathcal{C}}_{\mathcal{A}} = \mathcal{C}^1_{\mathcal{A}}$ and $\mathcal{C}_{\mathcal{A}}(t) = \mathcal{C}^1_{\mathcal{A}}$.

Theorem 4.1. *The stability region of the CRB scheduling policy is* Λ^{CRB}*, and*

$$\Lambda^{CRB} = \bigcup_{\sum_i t_i = 1} \sum_{i=1}^{|\mathbb{S}|} \mathbf{R}^{CRB}_{\mathcal{S}_i} t_i, \tag{4.7}$$

where $t_i \in \mathbb{R}_+ \bigcup \{0\}$*,*

$$\mathbf{R}^{CRB}_{\mathcal{S}_i} = \arg\max_{\mathbf{r} \in \bar{\mathcal{C}}_{\mathcal{S}_i}} \sum_{j \in \mathcal{S}_i} f(R^{CRB}_j) r_j, \tag{4.8}$$

and $\mathbf{R}^{CRB} = \mathbf{R}^{CRB}_{\mathcal{N}}$*.*

Proof. Since the scheduler is the CRB one, the update of $\mathbf{R}^{\mathrm{CRB}}(t)$ is independent of the scheduling decision in each time slot, and $\mathbf{R}^{\mathrm{CRB}}(t)$ converges to $\mathbf{R}^{\mathrm{CRB}}_{\mathcal{N}}$, *i.e.*, $\mathbf{R}^{\mathrm{CRB}} = \mathbf{R}^{\mathrm{CRB}}_{\mathcal{N}}$.

Comparing (4.6) with (4.8), we can conclude that

$$\mathbb{E}_t[\mathbf{r}^{\mathrm{CRB}}_{\mathcal{A}(t)}(t)] = \mathbb{E}_i[\mathbf{R}^{\mathrm{CRB}}_{\mathcal{S}_i}]$$

if $\mathbf{R}^{\mathrm{CRB}}(t)$ converges. This is because $\mathbf{R}^{\mathrm{CRB}}_{\mathcal{S}_i}$ is the average throughput of user set $\mathcal{S}_i$ over time, and by taking the expectation over i, $\mathbb{E}_i[\mathbf{R}^{\mathrm{CRB}}_{\mathcal{S}_i}]$ is the average throughput of the system. Since $\mathbf{r}^{\mathrm{CRB}}_{\mathcal{A}(t)}(t)$ is the throughput of the system in time slot t, by taking expectation over time, $\mathbb{E}_t[\mathbf{r}^{\mathrm{CRB}}_{\mathcal{A}(t)}(t)]$ is also the average throughput of the system.

If the system is stable, the average arrival rate should be equal to the average throughput, *i.e.*,

$$\boldsymbol{\lambda} = \mathbb{E}_t[\mathbf{r}^{\mathrm{CRB}}_{\mathcal{A}(t)}(t)] = \mathbb{E}_i[\mathbf{R}^{\mathrm{CRB}}_{\mathcal{S}_i}],$$

and consequently, the necessary condition for the system to be stable is that we can find a $\mathbf{t}$ ($\sum_i t_i = 1$) such that

$$\boldsymbol{\lambda} = \sum_{i=1}^{|\mathbb{S}|} \mathbf{R}^{\mathrm{CRB}}_{\mathcal{S}_i} t_i = \mathbb{E}_i[\mathbf{R}^{\mathrm{CRB}}_{\mathcal{S}_i}],$$

which is equivalent to $\boldsymbol{\lambda} \in \Lambda^{\mathrm{CRB}}$.

The sufficient condition can be proved by contradiction. Suppose $\boldsymbol{\lambda} \in \Lambda^{\mathrm{CRB}}$, but the system is not stable, and therefore at least one queue is unstable. Suppose that the queues in set $\mathcal{Q}$ are unstable, and the queues in set $\mathcal{N}|\mathcal{Q}$ are stable. Since queue $i \in \mathcal{Q}$ is unstable, we have $\mathbb{E}[q_i(t)] \to \infty$ which suggests that user i is always scheduled. Suppose that user set $\mathcal{D}$ is the scheduled user set, then we have $\mathcal{Q} \subseteq \mathcal{D}$. We further construct a set $\mathbb{D}$ which is made up of all $\mathcal{D}$. Therefore the average throughput of the system is

$$\bar{\mathbf{R}} = \sum_{\mathcal{D} \in \mathbb{D}} \pi_{\mathcal{D}} \mathbf{R}_{\mathcal{D}}^{\mathrm{CRB}},$$

and $\sum_{\mathcal{D} \in \mathbb{D}} \pi_{\mathcal{D}} = 1$. Because $\mathcal{D}$ is nonempty, we have

$$\bar{\mathbf{R}} \in \Lambda^{\mathrm{CRB}},$$

and for any $\boldsymbol{\epsilon}$, with $\sum_i \epsilon_i > 0$, $\epsilon_i \in \mathbb{R}_+ \bigcup \{0\}$,

$$\bar{\mathbf{R}} + \boldsymbol{\epsilon} \notin \Lambda^{\mathrm{CRB}}.$$

Due to the assumption of the stability of the system, we have

$$\begin{cases} \bar{R}_i < \lambda_i, \forall i \in \mathcal{Q}, \\ \bar{R}_i = \lambda_i, \forall i \in \mathcal{N}|\mathcal{Q}. \end{cases}$$

Consequently, there exists an $\boldsymbol{\epsilon}$ that $\bar{\mathbf{R}} + \boldsymbol{\epsilon} = \boldsymbol{\lambda} \in \Lambda^{\mathrm{CRB}}$, which is contradicted with (4.9). Thus the assumption cannot hold, and we have proved $\forall \boldsymbol{\lambda} \in \Lambda^{\mathrm{CRB}}$, the system is stable.

In summary, the stability region of the system is Λ^{CRB}. □

Here, due to the special property of the capacity region, the stability region equals the capacity region. Note that the capacity region is a Euclidean simplex with $N+1$ vertices and each vertex represents a rate vector. Suppose the $N+1$ vertices make up a set $\mathcal{V}$. Since $\mathbf{R}_{\mathcal{S}_i}$ is on the boundary of the capacity region $\mathcal{C}_{\mathcal{S}_i}$, it lies in the hyperplane determined by the points in $\mathcal{V}$. The stability region is the convex hull of $\mathbf{R}_{\mathcal{S}_i}$, which equals the convex hull of $\mathcal{V}$, *i.e.*, the capacity region.

4.3.2 Stochastic Channel Case

For the stochastic channel case, we have a similar result to the static channel case.

Theorem 4.2. *Theorem 4.1 holds for the stochastic channel case.*

Proof. Similar to the static channel case, $\mathbf{R}^{\mathrm{CRB}}(t)$ converges to

$$\mathbf{R}^{\mathrm{CRB}} = \arg\max_{\mathbf{r}\in\bar{\mathcal{C}}_{\mathcal{N}}} \sum_{j\in\mathcal{N}} U(r_j),$$

and we have

$$\mathbf{r}_{\mathcal{S}_i}^{\mathrm{CRB}}(t) = \arg\max_{\mathbf{r}\in\mathcal{C}_{\mathcal{S}_i}(t)} \sum_{j} f(R_j^{\mathrm{CRB}}) r_j.$$

Taking expectation over time, we have

$$\mathbf{R}_{\mathcal{S}_i}^{\mathrm{CRB}} = \arg\max_{\mathbf{r}\in\bar{\mathcal{C}}_{\mathcal{S}_i}} \sum_{j} f(R_j^{\mathrm{CRB}}) r_j.$$

Also as

$$\mathbf{R}^{\mathrm{CRB}} = \mathbf{R}_{\mathcal{N}}^{\mathrm{CRB}},$$

and based on the same approach as in the static channel case, we can prove that Theorem 4.1 holds for the stochastic channel case. □

Worth to note that, different from the static channel case where the stability region is identical to the capacity region, the stability region in the stochastic channel case is generally less than the capacity region due to the fact that the ergodic capacity region is a convex polytope, but not necessarily a Euclidean simplex.

4.4 Stability Region of the UB scheduling

Similar to the discussion of the CRB scheduling, we first discuss the simple case, the static channel case, and then study the complicated stochastic channel case. Furthermore, we show that, the results obtained in the static channel case can be directly used in the stochastic channel case, by replacing the capacity region with the ergodic one. Different from the CRB scheduling, whose stability region can be easily

obtained in closed-form, the stability region of the UB scheduling generally cannot be obtained in closed-form, and therefore we develop a numerical method to tackle the two-user case.

4.4.1 Static Channel Case

According to [43], $\mathbf{R}^{\mathrm{UB}}(t)$ converges to the average throughput. Once $\mathbf{R}^{\mathrm{UB}}(t)$ converges, we have

$$\mathbf{R}_{\mathcal{S}_i}^{\mathrm{UB}} = \arg\max_{\mathbf{r}\in\bar{\mathcal{C}}_{\mathcal{S}_i}} \sum_j f(R_j^{\mathrm{UB}}) r_j,$$

and

$$\mathbf{R}^{\mathrm{UB}} = \mathbb{E}_i[\mathbf{R}_{\mathcal{S}_i}^{\mathrm{UB}}].$$

Note that generally

$$\mathbf{R}^{\mathrm{UB}} \neq \arg\max_{\mathbf{r}\in\bar{\mathcal{C}}_{\mathcal{N}}} \sum_{i\in\mathcal{N}} U(r_i),$$

and $\mathbf{R}^{\mathrm{UB}}$ may not lie on the boundary of the capacity region.

We have the following theorem to verify whether a system with a specific arrival rate vector is stable or not.

Theorem 4.3. *A system using the UB scheduling policy with average arrival rate* $\boldsymbol{\lambda}$ *is stable if and only if* $\boldsymbol{\lambda} \in \tilde{\Lambda}^{UB}(\boldsymbol{\lambda})$*, where*

$$\tilde{\Lambda}^{UB}(\boldsymbol{\lambda}) = \bigcup_{\sum_i t_i = 1} \sum_{i=1}^{|\mathbb{S}|} \mathbf{R}_{\mathcal{S}_i}^{UB} t_i,$$

and

$$\mathbf{R}_{\mathcal{S}_i}^{UB} = \arg\max_{\mathbf{r}\in\bar{\mathcal{C}}_{\mathcal{S}_i}} \sum_j f(\lambda_j) r_j.$$

Proof. Suppose that the system is stable, and then we have $\boldsymbol{\lambda} = \mathbf{R}^{\mathrm{UB}}$. Thus, the

average rate allocated to user set $\mathcal{S}_i$ is

$$\mathbf{R}_{\mathcal{S}_i}^{\mathrm{UB}} = \arg\max_{\mathbf{r}\in\bar{\mathcal{C}}_{\mathcal{S}_i}} \sum_i f(R_i^{\mathrm{UB}}) r_i = \arg\max_{\mathbf{r}\in\bar{\mathcal{C}}_{\mathcal{S}_i}} \sum_i f(\lambda_i) r_i,$$

if $\mathcal{S}_i$ is scheduled. Since $\mathbf{R}^{\mathrm{UB}} = \mathbb{E}_i[\mathbf{R}_{\mathcal{S}_i}^{\mathrm{UB}}]$, we have $\boldsymbol{\lambda} = \mathbb{E}_i[\mathbf{R}_{\mathcal{S}_i}^{\mathrm{UB}}] \in \tilde{\Lambda}^{\mathrm{UB}}(\boldsymbol{\lambda})$.

Based on the same argument as that in Sec. 4.3, we can prove that $\forall \boldsymbol{\lambda} \in \Lambda^{\mathrm{UB}}$, the system is stable. Thus, the theorem is proved. □

Based on the above theorem, we have the following corollary.

Corollary 4.4. *If* $\boldsymbol{\lambda} = \arg\max_{\mathbf{r}\in\bar{\mathcal{C}}_{\mathcal{N}}} U(r_i)$*, then the system is stable.*

Proof. Since

$$\boldsymbol{\lambda} = \arg\max_{\mathbf{r}\in\bar{\mathcal{C}}_{\mathcal{N}}} U(r_i), \tag{4.9}$$

we have

$$\boldsymbol{\lambda} = \mathbf{R}_{\mathcal{N}}^{\mathrm{UB}} = \arg\max_{\mathbf{r}\in\bar{\mathcal{C}}_{\mathcal{N}}} \sum_{i\in\mathcal{N}} f(\lambda_i) r_i,$$

which means $\boldsymbol{\lambda} \in \tilde{\Lambda}^{\mathrm{UB}}(\boldsymbol{\lambda})$, and thus the system is stable. □

The corollary states that at least one point on the outer-bound[1] of the capacity region can be stabilized by the UB scheduling.

Based on Theorem 4.3, we have the following theorem to quantify the stability region of the UB scheduling.

Theorem 4.5. *The stability region of the UB scheduling policy is* Λ^{UB}*, and for any* $\boldsymbol{\lambda} \in \Lambda^{UB}$*, Theorem 4.3 holds; for any* $\boldsymbol{\lambda} \notin \Lambda^{UB}$*, Theorem 4.3 does not hold.*

Proof. The theorem can be directly obtained based on the definition of the stability region and Theorem 4.3. □

Similar to the CRB scheduling in the static channel case, the stability region of the UB scheduling also equals to the capacity region in the static channel case.

[1] A point lies on the outer-bound of a set should satisfy two conditions: first is that the point lies on the boundary of a set; second is that the point is no longer belongs to the set if any increment in any dimension is made to the point.

4.4.2 Stochastic Channel Case

Theorem 4.6. *Theorem 4.3 and Theorem 4.5 hold for the stochastic channel case.*

Proof. Similar to the static channel case, $\mathbf{R}^{\mathrm{UB}}(t)$ converges to the average throughput of the system. Then we have

$$\mathbf{r}_{\mathcal{S}_i}^{\mathrm{UB}}(t) = \arg\max_{\mathbf{r}\in\mathcal{C}_{\mathcal{S}_i}(t)} \sum_i f(R_i^{\mathrm{UB}})r_i.$$

Taking expectation over time, we have

$$\mathbf{R}_{\mathcal{S}_i}^{\mathrm{UB}} = \arg\max_{\mathbf{r}\in\bar{\mathcal{C}}_{\mathcal{S}_i}} \sum_i f(R_i^{\mathrm{UB}})r_i,$$

and $\mathbf{R}^{\mathrm{UB}} = \mathbb{E}_i[\mathbf{R}_{\mathcal{S}_i}]$. Then we can follow the same approach as in the static channel case. By replacing the static capacity region with the ergodic capacity region, the discussions in the static channel case also hold for the stochastic channel case. Therefore we can prove that Theorem 4.3 holds for the stochastic channel case. Then based on the definition of the stability region, we can show that Theorem 4.5 holds for the stochastic channel case. □

While different from the static channel case, where the stability region can be obtained in closed-form, the stability region in the stochastic channel case is hard to be derived in closed-form. But we discover two properties as follows.

Proposition 4.7. *The stability region of the UB scheduling policy can be non-convex.*

Proposition 4.8. *With the UB scheduler, even though the system is stable when the arrival rate is $\boldsymbol{\lambda}$, the system can be unstable when the arrival rate is reduced to $x\boldsymbol{\lambda}$, where $0 < x < 1$.*

For these two properties, we only need to show that they hold for some scheduling policies with specific function f. This will be done in the Sec. 4.6.

Remark. These two properties make the UB scheduling very undesirable if the function f is selected improperly. The non-convexity means if one user decreases its arrival rate, the system may be unstable which is harmful for the quality of service. The second property means that reducing the traffic intensity may bring a stable system to an unstable system which will also damage the QoS for all on-going traffic.

Although the closed-form expression of the stability region is difficult to obtain, a numerical method can be used to obtain the stability region. Here, we give the method to obtain the stability region of two-user systems, and it can be easily extended to a more general case.

Numerical method to obtain the stability region of two-user systems

Since the ergodic capacity region is a compact, convex, coordinate convex polyhedron, it can be represented as

$$\mathcal{C} = \{(r_1, r_2) : a_k r_1 + r_2 \leq b_k, k = 1, 2, ..., K\},$$

where a_k is in the increasing order w.r.t. k, and if $a_k = \infty$, then the corresponding equation is $r_1 = b_k$.

Let $\mathbf{r}^k = (r_1^k, r_2^k)$ be the solution of

$$\begin{cases} a_k r_1 + r_2 = b_k, \\ a_{k+1} r_1 + r_2 = b_{k+1}, \end{cases}$$

where $0 < k < K$, $\mathbf{r}^0$ be the solution of

$$\begin{cases} a_1 r_1 + r_2 = b_1, \\ r_1 = 0, \end{cases}$$

and $\mathbf{r}^K$ be the solution of

$$\begin{cases} a_K r_1 + r_2 = b_K, \\ r_2 = 0. \end{cases}$$

Geometrically, $\mathbf{r}^k$ is the vertex on the outer bound of the capacity region.

If $f(\lambda_1)/f(\lambda_2) \in (a_k, a_{k+1})$, *i.e.*,

$$\lambda \in Z_k = \{(\lambda_1, \lambda_2) : f(\lambda_1)/f(\lambda_2) \in (a_k, a_{k+1})\},$$

the stability region is the convex hull of $\{\mathbf{0}, \mathbf{r}^0, \mathbf{r}^k, \mathbf{r}^K\}$, which is represented as

$$\begin{aligned} \Lambda_k = \quad & \{(r_1, r_2) = \beta_1 \mathbf{r}^0 + \beta_2 \mathbf{r}^k + \beta_3 \mathbf{r}^K : \\ & \forall i, \beta_i > 0, \textstyle\sum_i \beta_i \leq 1\}. \end{aligned}$$

If $f(\lambda_1)/f(\lambda_2) = a_k$, then the stability region is

$$\Lambda^k = \{(r_1, r_2) : a_k r_1 + r_2 \leq b_k, f(r_1)/f(r_2) = a_k\}.$$

Overall, the stability region can be represented as

$$\Lambda^{\text{UB}} = \bigcup_k (\Lambda_k \bigcap Z^k) \bigcup \Lambda^k.$$

Remark. The key idea of the numerical method is to partition the capacity region into zones (Z_k) and partition curves (Λ^k). Each partition curve is the curve along the boundary of two neighboring zones. Since the capacity region is a convex polyhedron, the number of zones is finite[2]. For each zone, the allocated rate is identical, and thus the stability region for the arrival rate in each zone can be obtained. Examples are given in Sec. 4.6 to show how to use the proposed method to obtain the stability region.

4.5 Extended Stability Region

4.5.1 Extended Stability Region of the CRB Scheduling

If we give a weight to each user, then a more general CRB scheduling policy is to allocate the rate based on the following optimization problem if user set $\mathcal{A}(t)$ is selected:

$$\mathbf{r}^{\mathbf{w},\text{CRB}}_{\mathcal{A}(t)}(t) = \arg\max_{\boldsymbol{\eta} \in \mathcal{C}_{\mathcal{A}(t)}(t)} \sum_{i \in \mathcal{A}(t)} w_i f(R_i^{\text{CRB}}(t)) \eta_i,$$

where $w_i \in \mathbb{R}_+ \bigcup \{0\}$ is the normalized weight, satisfying $\sum_i w_i = 1$.

Since with the CRB scheduler, $\mathbf{R}^{\text{CRB}}(t)$ converges to

$$\mathbf{R}^{\mathbf{w},\text{CRB}} = \arg\max_{\mathbf{r} \in \bar{\mathcal{C}}_{\mathcal{N}}} \sum_{i \in \mathcal{N}} w_i U(r_i). \tag{4.10}$$

[2]Note that if the outer bound of the capacity region is strict convex, then the number of zones is infinite, and this method cannot work.

Similar to (4.8), we have

$$\mathbf{R}_{\mathcal{S}_i}^{\mathbf{w},\mathrm{CRB}} = \arg\max_{\mathbf{r}\in\bar{\mathcal{C}}_{\mathcal{S}_i}} \sum_{j\in\mathcal{S}_i} w_j f(R_j^{\mathbf{w},\mathrm{CRB}}) r_j,$$

and $\mathbf{R}^{\mathbf{w},\mathrm{CRB}} = \mathbf{R}_{\mathcal{N}}^{\mathbf{w},\mathrm{CRB}}$.

Similar to (4.7), for any given weight $\mathbf{w}$, the corresponding stability region is obtained as

$$\Lambda^{\mathbf{w},\mathrm{CRB}} = \bigcup_{\sum_i t_i=1} \sum_{i=1}^{|\mathbb{S}|} \mathbf{R}_{\mathcal{S}_i}^{\mathbf{w},\mathrm{CRB}} t_i,$$

where $t_i \in \mathbb{R}_+ \bigcup\{0\}$, and we have the following theorem.

Theorem 4.9.

$$\bar{\mathcal{C}}_{\mathcal{N}} = \bigcup_{\sum_i w_i=1} \Lambda^{\mathbf{w},CRB},$$

where $\Lambda^{\mathbf{w},CRB}$ *is the stability region of the CRB scheduling with weight* $\mathbf{w}$ *assigning to users.*

Proof. In order to prove that the union of the weighted stability region is the ergodic capacity region, essentially we need to show that, all the boundary points of the capacity region are the solutions to (4.10) by varying the weight $\mathbf{w}$.

By mapping $\bar{\mathcal{C}}_{\mathcal{N}}$ to $\bar{\mathcal{C}}_{\mathcal{N}}^U$ through $U(x)$, *i.e.*,

$$\bar{\mathcal{C}}_{\mathcal{N}}^U = \{(U(x_1), U(x_2), ..., U(x_N))|\mathbf{x} \in \bar{\mathcal{C}}_{\mathcal{N}}\},$$

(4.10) can be represented as

$$\mathbf{y}^{\mathbf{w},\mathrm{CRB}} = \arg\max_{\mathbf{y}\in\bar{\mathcal{C}}_{\mathcal{N}}^U} \sum_{i\in\mathcal{N}} w_i y_i. \tag{4.11}$$

If $\bar{\mathcal{C}}_{\mathcal{N}}^U$ is a closed convex set, then according to the supporting hyperplane theorem [7], for any point $\mathbf{y}$ lies on the boundary of $\bar{\mathcal{C}}_{\mathcal{N}}^U$, we can find the corresponding $\mathbf{w}$ such that $\mathbf{y} = \mathbf{y}^{\mathbf{w},\mathrm{CRB}}$. Thus all the boundary points of $\bar{\mathcal{C}}_{\mathcal{N}}^U$ are the solutions to (4.11) by varying $\mathbf{w}$. Since $U(x)$ is a monotonic non-decreasing function, the images of the boundary points of $\bar{\mathcal{C}}_{\mathcal{N}}^U$ in $\bar{\mathcal{C}}_{\mathcal{N}}$ are still the boundary points. Then, the theorem can

be proved if $\bar{\mathcal{C}}_{\mathcal{N}}^{U}$ is a closed convex set.

For any $\mathbf{x^m}, \mathbf{x^n} \in \bar{\mathcal{C}}_{\mathcal{N}}$ whose image in $\bar{\mathcal{C}}_{\mathcal{N}}^{U}$ is $\mathbf{m}, \mathbf{n}$, respectively, we have

$$\begin{aligned}&\alpha\mathbf{m} + (1-\alpha)\mathbf{n}\\ =&\alpha(U(x_1^{\mathbf{m}}), ..., U(x_N^{\mathbf{m}})) + (1-\alpha)(U(x_1^{\mathbf{n}}), ..., U(x_N^{\mathbf{n}}))\\ =&(y_1, ..., y_N),\end{aligned}$$

where

$$y_i = \alpha U(x_i^{\mathbf{m}}) + (1-\alpha)U(x_i^{\mathbf{n}}).$$

Without loss of generality, we assume $x_i^{\mathbf{m}} \geq x_i^{\mathbf{n}}$. Because $U(x)$ is monotonically non-decreasing, $y_i \in [U(x_i^{\mathbf{n}}) \quad U(x_i^{\mathbf{m}})]$. *i.e.*, $x_i = U^{-1}(y_i) \in [x_i^{\mathbf{n}} \quad x_i^{\mathbf{m}}]$. Consequently, $\mathbf{x} \in \bar{\mathcal{C}}_{\mathcal{N}}$ due to the convexity of $\bar{\mathcal{C}}_{\mathcal{N}}$, and therefore $\mathbf{y} \in \bar{\mathcal{C}}_{\mathcal{N}}^{U}$ and the convexity of $\bar{\mathcal{C}}_{\mathcal{N}}^{U}$ is proved. Since $\bar{\mathcal{C}}_{\mathcal{N}}$ is closed, $\bar{\mathcal{C}}_{\mathcal{N}}^{U}$ is also closed. Therefore $\bar{\mathcal{C}}_{\mathcal{N}}^{U}$ is indeed a closed convex set. □

4.5.2 Extended Stability Region of the UB Scheduling

Similar to the CRB scheduling, for the UB scheduling, a slight modification to the scheduling policy by giving a weight to each user, the resultant stability region is denoted by $\Lambda^{\mathbf{w},\mathrm{UB}}$, and we have the following theorem.

Theorem 4.10.

$$\bar{\mathcal{C}}_{\mathcal{N}} = \bigcup_{\sum_i w_i = 1} \Lambda^{\mathbf{w},UB},$$

where $\Lambda^{\mathbf{w},UB}$ is the stability region of the UB scheduling with weight $\mathbf{w}$ assigning to users.

Proof. First, based on the supporting hyperplane theorem, for any given $\boldsymbol{\lambda}$ and any boundary point of $\bar{\mathcal{C}}_{\mathcal{S}_i}$, we can find a $\mathbf{w}$ such that the boundary point $\mathbf{R}_{\mathcal{S}_i}^{\mathbf{w},\mathrm{UB}}$ is the solution to the following problem,

$$\mathbf{R}_{\mathcal{S}_i}^{\mathbf{w},\mathrm{UB}} = \arg\max_{\mathbf{r} \in \bar{\mathcal{C}}_{\mathcal{S}_i}} \sum_{j \in \mathcal{S}_i} w_j f(\lambda_j) r_j,$$

where $\sum_j w_j = 1$. Also note that the boundary points of $\bar{\mathcal{C}}_{\mathcal{S}_i}$ also lie on the boundary of $\bar{\mathcal{C}}_{\mathcal{N}}$; therefore, we have

$$\begin{aligned}\bar{\mathcal{C}}_{\mathcal{N}} &= \bigcup_{\sum_i w_i=1} \mathbf{Co}\{\mathbf{R}_{\mathcal{S}_i}^{\mathbf{w},\mathrm{UB}} : i \in \mathcal{N}\}, \\ &= \bigcup_{\sum_i w_i=1} \tilde{\Lambda}^{\mathbf{w},\mathrm{UB}}(\boldsymbol{\lambda})\end{aligned}$$

where **Co** means convex hull, and

$$\tilde{\Lambda}^{\mathbf{w},\mathrm{UB}}(\boldsymbol{\lambda}) = \bigcup_{\sum_i t_i=1} \sum_{i=1}^{|\mathbb{S}|} \mathbf{R}_{\mathcal{S}_i}^{\mathrm{UB}} t_i.$$

According to Theorem 4.5, we know that

$$\boldsymbol{\lambda} \in \Lambda^{\mathbf{w},\mathrm{UB}} \Leftrightarrow \boldsymbol{\lambda} \in \tilde{\Lambda}^{\mathbf{w},\mathrm{UB}}(\boldsymbol{\lambda}).$$

So we have

$$\boldsymbol{\lambda} \in \bigcup_{\sum_i w_i=1} \Lambda^{\mathbf{w},\mathrm{UB}} \Leftrightarrow \boldsymbol{\lambda} \in \bigcup_{\sum_i w_i=1} \tilde{\Lambda}^{\mathbf{w},\mathrm{UB}}(\boldsymbol{\lambda}) = \bar{\mathcal{C}}_{\mathcal{N}},$$

which suggests

$$\bigcup_{\sum_i w_i=1} \Lambda^{\mathbf{w},\mathrm{UB}} = \bar{\mathcal{C}}_{\mathcal{N}}.$$

□

4.5.3 Discussion

Although the stability regions of the CRB and the UB scheduling policies are less than the capacity region, respectively, by assigning the weights to users, the resultant scheduling algorithms can stabilize the system. Further note that, by giving the weights to users, the equivalent utility function has changed from a homogeneous one ($U(.)$) to a heterogeneous one ($w_i U(.)$). Therefore, for any given $\mathbf{w}$, the discussion in Sec. 4.3 and Sec. 4.4 can still be used to analyze the stability of the system.

The advantage of the weighted opportunistic scheduling is that when the arrival

rate lies outside the capacity region, the operation point (the throughput) is determined by the utility function U (in both UB and CRB scheduling policies), which is typically designed based on the fairness concern. Therefore the weighted opportunistic scheduling can provide a better fairness.

Although the approach is promising, it may not be easy. The weight-design is to find the supporting hyperplane (weight) of a closed convex set (capacity region) in a specific boundary point (the intersection of the arrival rate vector and the capacity region). Since the solution highly depends on the shape of the closed convex set, we lack a general analytic method. Further work should be done to obtain a simple method to design the weight.

4.6 Examples and Sample Validation

In this section, we give examples about the stability region of the UB scheduling and the CRB scheduling policies. Simulation is conducted to compare the two policies, and validate the analytic results.

4.6.1 Channel Assumption

Considering a two-user four-state channel, the transmission rate vector is

$$\mathbf{u}^m = \begin{cases} [R_1^{\mathrm{ON}}, 0]^T, & m = 1, \\ [0, R_2^{\mathrm{ON}}]^T, & m = 2, \\ [R_1^{\mathrm{ON}}, R_2^{\mathrm{ON}}]^T, & m = 3, \\ [0, 0]^T, & m = 4, \end{cases}$$

and the stationary distribution is $\boldsymbol{\pi} = (1/4, 1/4, 1/4, 1/4)$. Note that this is a channel model for a two-user system, where each user has two states (ON and OFF), and the channel states for different users are independent. The achievable throughput of user i is R_i^{ON} when its channel state is ON, and 0 if its channel state is OFF. Without loss of generality, we assume $R_1^{\mathrm{ON}} \geq R_2^{\mathrm{ON}}$.

Based on (4.1), we can obtain the ergodic capacity region as:

$$\begin{aligned} \bar{\mathcal{C}} \ &= \{(R_1, R_2) : R_1/R_1^{\mathrm{ON}} + R_2/R_2^{\mathrm{ON}} \leq 3/4, \\ &\qquad R_1/R_1^{\mathrm{ON}} \leq 1/2, R_2/R_2^{\mathrm{ON}} \leq 1/2\}. \end{aligned}$$

4.6.2 Utility Function

α-Fairness Utility

The utility function chosen to be evaluated is the α-fairness ones [58]:

$$U(x) = \begin{cases} \log(x), & \alpha = 1, \\ (1-\alpha)^{-1}x^{1-\alpha}, & \text{otherwise}, \end{cases}$$

where x is the average throughput, whose unit is bps/Hz and is omitted in the following. The derivative of $U(x)$ is

$$f(x) = x^{-\alpha}.$$

By choosing different α, the objective is to maximize the fairness measurement based on different principles, and the relative value of the measurement is of more interests. For instance, if $\alpha = 0$, then the objective is to maximize the system throughput; if $\alpha = 1$, then it is to maximize the proportional fairness; if $\alpha \to \infty$, then it is to maximize the max-min fairness.

Exponential Utility

Another utility function chosen to be evaluated is exponential utility [13]:

$$U(x) = -\frac{1}{a}e^{-ax},$$

and

$$f(x) = e^{-ax}.$$

For the exponential utility, the marginal utility is exponentially decreasing, and the changing rate of the marginal utility is a constant and independent of x.

4.6.3 Stability Region of the UB Scheduling

α-Fairness Utility

Based on the numerical method proposed in Sec. 4.4, we can obtain the stability region, as shown in Fig. 4.1, Fig. 4.2 and Fig. 4.3.

The point P in the figure is the intersection of the boundary of the capacity region and curve $\frac{f(\lambda_1)}{f(\lambda_2)} = \beta$ where $\beta = R_2^{\mathrm{ON}}/R_1^{\mathrm{ON}}$. With the increasing of α, P moves along the boundary of the capacity region, and results in the shape changing of the stability region. From the figure we also can observe that the stability region is non-convex all the time. When the value of α is proper, the stability region is the union of a convex set and a line segment. When α is large or small, P moves to the line $R_1 = R_1^{\mathrm{ON}}/2$ or $R_2 = R_2^{\mathrm{ON}}/2$, then the stability region is a trapezoid minus a triangular. The non-convex property of the stability region makes the system behavior hard to predict and the QoS hard to meet, since decreasing the arrival rate of one flow may lead the system changing from stable to unstable.

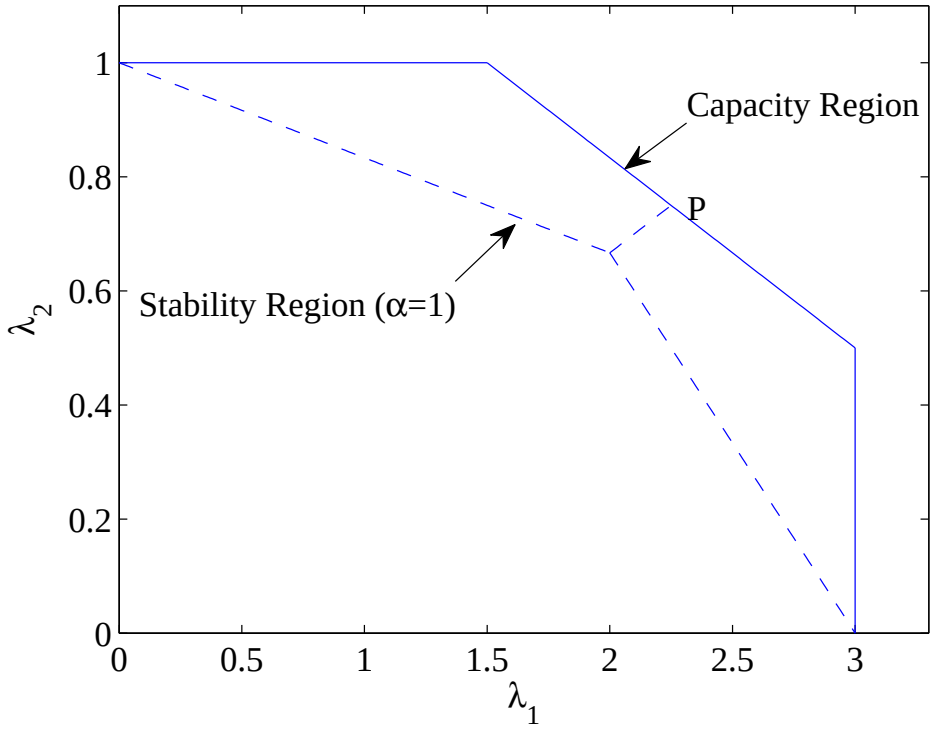


Figure 4.1: Stability Region of a system with four-state channel and α-fairness UB scheduling, $R_1^{\mathrm{ON}} = 6$, $R_2^{\mathrm{ON}} = 2$, $\alpha = 1$

Exponential Utility

We already know that, with the UB scheduler, for a stable system with arrival rate $\boldsymbol{\lambda}$, decreasing any element in $\boldsymbol{\lambda}$ may lead the system to be unstable. Here we give another example to show that, proportionally decreasing all the elements in $\boldsymbol{\lambda}$ (down-scale $\boldsymbol{\lambda}$)

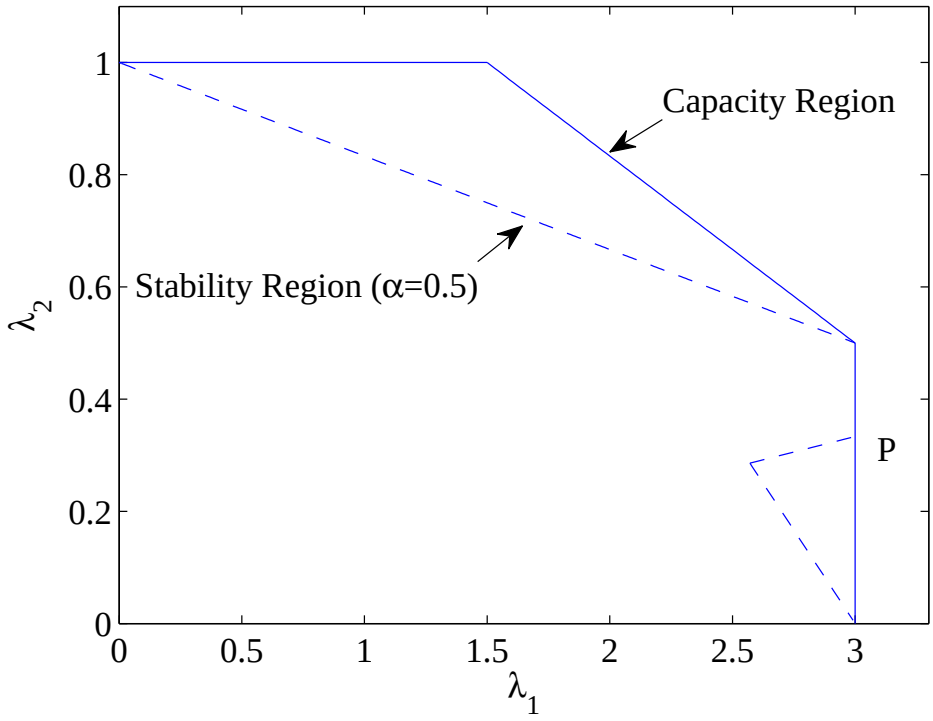


Figure 4.2: Stability Region of a system with four-state channel and α-fairness UB scheduling, $R_1^{\mathrm{ON}} = 6$, $R_2^{\mathrm{ON}} = 2$, $\alpha = 0.5$

may also lead the system to be unstable.

Based on the same approach as in α-fairness utility, the stability region can be obtained. Since we change the function f, so the curve $\frac{f(\lambda_1)}{f(\lambda_2)} = \beta$, which determines P, is $\lambda_2 - \lambda_1 = 1/a \log \beta$. If the arrival rate $\boldsymbol{\lambda}$ is downscaled by x, the new arrival rate is no longer lies in the partition curve $\frac{f(\lambda_1)}{f(\lambda_2)} = \beta$. Therefore the stability cannot be guaranteed, and the stability property should be examined by finding which zone the new $\boldsymbol{\lambda}$ lies in. As illustrated in Fig. 4.4 and Fig. 4.6, when $a = 1$ or 3, if the system is stable at point P, then 'down-scale' $\boldsymbol{\lambda}$ by x, the system becomes unstable, *i.e.*, suffering the 'down-scale' unstable; but as illustrated in Fig. 4.5, when $a = 0.4$, the 'down-scale' unstable situation does not happen.

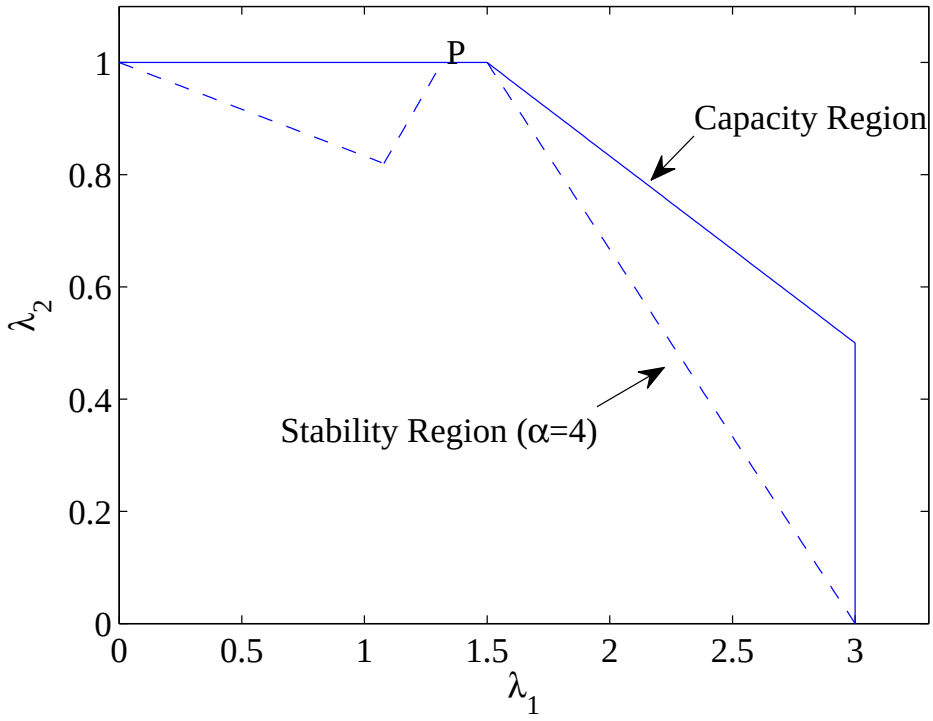


Figure 4.3: Stability Region of a system with four-state channel and α-fairness UB scheduling, $R_1^{\mathrm{ON}} = 6$, $R_2^{\mathrm{ON}} = 2$, $\alpha = 4$

4.6.4 Stability Region of the CRB Scheduling

α-Fairness Utility

We enumerate $\mathbb{S}$ as $\mathbb{S} = \{(1), (2), (1,2), \emptyset\}$. For each $\mathcal{A} \in \mathbb{S}$, we have $\mathbf{R}_{(1)} = [R_1^{\mathrm{ON}}/2, 0]^T$, $\mathbf{R}_{(2)} = [0, R_2^{\mathrm{ON}}/2]^T$, $\mathbf{R}_{\emptyset} = [0,0]^T$, and

$$\mathbf{R}_{(1,2)} = \begin{cases} [R_1^{\mathrm{ON}}/2, R_2^{\mathrm{ON}}/4]^T, & \alpha < \alpha_l, \\ [\frac{3R_1^{\mathrm{ON}}/4}{1+\beta^{1/\alpha-1}}, \frac{3R_2^{\mathrm{ON}}/4}{1+\beta^{1-1/\alpha}}]^T, & \alpha_l \le \alpha \le \alpha_h, \\ [R_1^{\mathrm{ON}}/4, R_2^{\mathrm{ON}}/2]^T, & \alpha > \alpha_h, \end{cases}$$

where $\alpha_l = \log\beta / \log\frac{\beta}{2}$, $\alpha_h = \log\beta / \log 2\beta$.

The capacity region and the stability region are illustrated in Fig. 4.7. By varying α, $\mathbf{R}_{(1,2)}$ is moving on the outer bound of the capacity region, and the stability region is always a convex hull.

Comparing Fig. 4.1, Fig. 4.2, Fig. 4.2 with Fig. 4.7, under the four-state channel assumption in a two-user system, the CRB scheduling policy can always provide a larger stability region than the UB scheduling policy if using the same utility function.

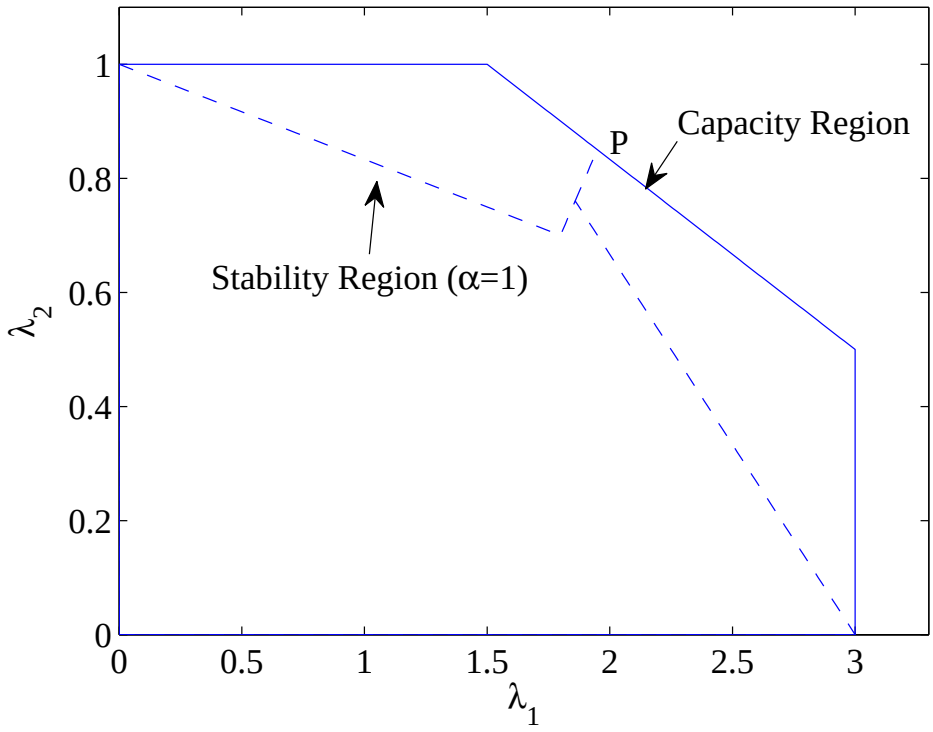


Figure 4.4: Stability Region of a system with four-state channel and exponential UB scheduling, $R_1^{\mathrm{ON}} = 6$, $R_2^{\mathrm{ON}} = 2$, $a = 1$.

4.6.5 Scheduling Policy Comparison

We have conducted simulation to compare the UB and the CRB scheduling policies. We choose α-fairness as the utility function, and $\alpha = 0.5$. We use Poisson traffic as the arrival traffic, ϵ is chosen as 0.01, and we run simulation 10 times to take the average. We set $\lambda_1 = R_1^{\mathrm{ON}}/2$ for all 20000 time slots, set $\lambda_2 = R_2^{\mathrm{ON}}/4$ for the first 10000 time slots and $\lambda_2 = R_2^{\mathrm{ON}}/10$ for the second 10000 time slots. This is used to simulate the arrival-rate decreasing of one flow.

The throughput comparison is shown in Fig. 4.8. From the figure, after 10000 time slots, the throughput of Q1 with the UB scheduler (the curve UB Q1) starts to decrease and is less than the throughput of Q1 with the CRB scheduler. For the throughput of Q2, both schedulers can maintain the same throughput, which equals the arrival rate of the second flow. Here we can conclude, by decreasing the arrival rate of one flow, the throughput of another flow can be decreased, if the UB scheduler is used. This phenomenon can be explained by examining the system stability based on Theorem 4.6 with the new arrival rate. An intuitive explanation is as follows: as the utility function $U(x)$ is strictly concave, the derivative function $f(x)$

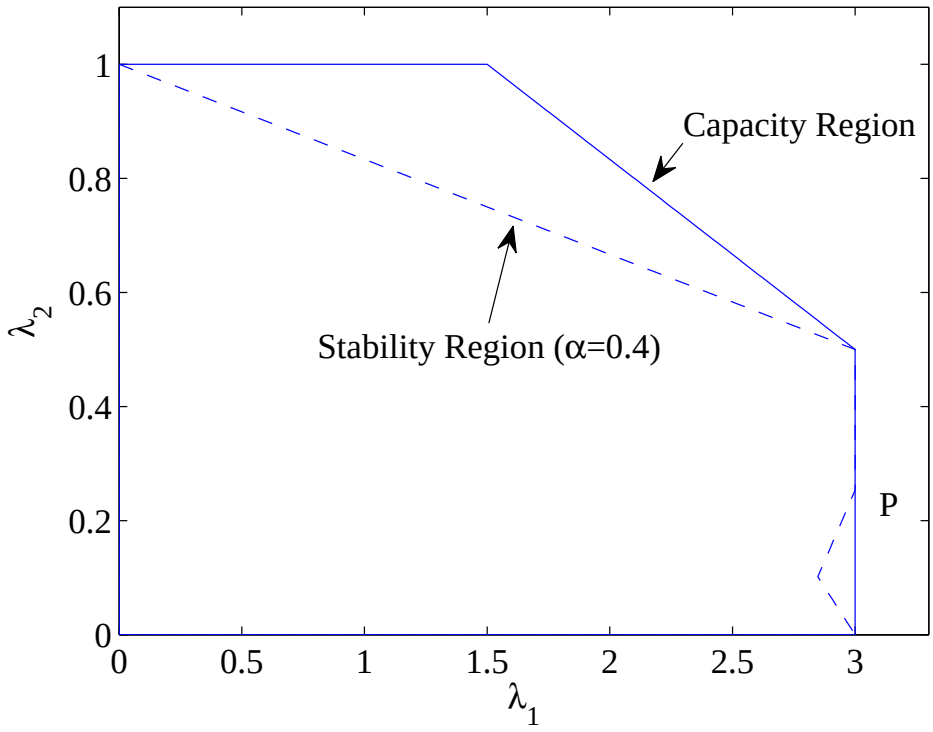


Figure 4.5: Stability Region of a system with four-state channel and exponential UB scheduling, $R_1^{\mathrm{ON}} = 6$, $R_2^{\mathrm{ON}} = 2$, $a = 0.4$

is a decreasing function. Since $R_i^{\mathrm{UB}}(t)$ is used to estimate the average throughput of user i and if the arrival rate of user i decreases, the estimated average throughput should also decrease, *i.e.*, $R_i^{\mathrm{UB}}(t)$ decreases. Therefore, $f(R_i^{\mathrm{UB}}(t))$ will increase. From (4.5) we can see, this generally results in the increase of $r_i^{\mathrm{UB}}(t)$, *i.e.*, the increase of the instantaneous rate of user i. The probability that the system stays in a state without user i will increase, as a joint results of the decreasing of the arrival rate and the increasing of the instantaneous rate. As the number of users has decreased, the system will lose certain multi-user diversity, *i.e.*, the achievable rate region will shrink. This may lead to a situation that the average throughput of a user except i is less than its arrival rate, and therefore leads to an unstable flow.

If we give weights to users[3], we can see the weighted UB scheduler can maintain the throughputs for both users. But further note that, although the specific weighted UB scheduling can stabilize the system with the arrival-rate decreasing of one flow in the given scenario, there will exist some scenarios that the system still cannot be

[3]Here we assign weight 0.75 to user 1 and 0.25 to user 2, and the corresponding curves are UB^{w} Q1 and UB^{w} Q2. The weight is specifically designed in order to stabilize the system, and such design is also non-unique.

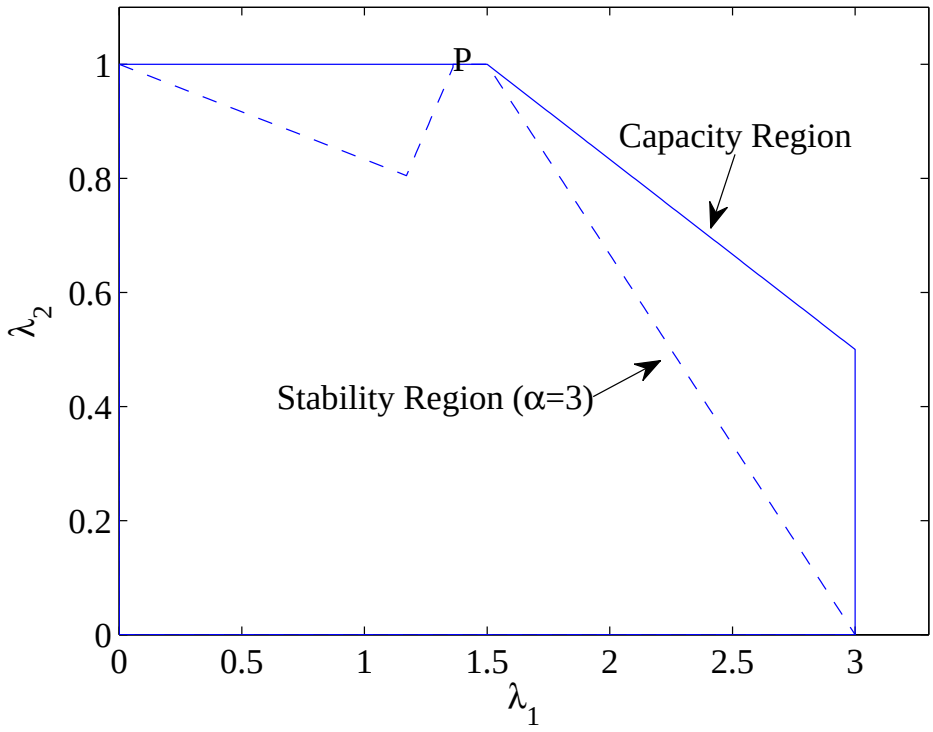


Figure 4.6: Stability Region of a system with four-state channel and exponential UB scheduling, $R_1^{\mathrm{ON}} = 6$, $R_2^{\mathrm{ON}} = 2$, $a = 3$.

stabilized if one flow decreases its arrival rate, as the stability region of the weighted UB scheduling is still less than the capacity region.

The queue length is compared in Fig. 4.9. The y-axis is in the logarithm form. After 10000 time slots, while the arrival rate of Q2 is reduced, with the UB scheduler, the queue length of Q1 starts to increase. From the increasing trend we could tell that, the system cannot be stabilized. But with a proper weight assigning to each user, the system can be stabilized by the weighted UB scheduling policy. These results validate our analytical conclusion.

4.7 Discussion and Conclusion

In this chapter, the stability regions of two opportunistic scheduling policies have been discussed. One is the UB scheduling policy and the other is a variant of the UB scheduling policy, called the CRB scheduling policy. We have proposed a numerical method to obtain the stability region of the UB scheduling policy, and the results show that the stability region of the UB scheduling policy is generally non-convex

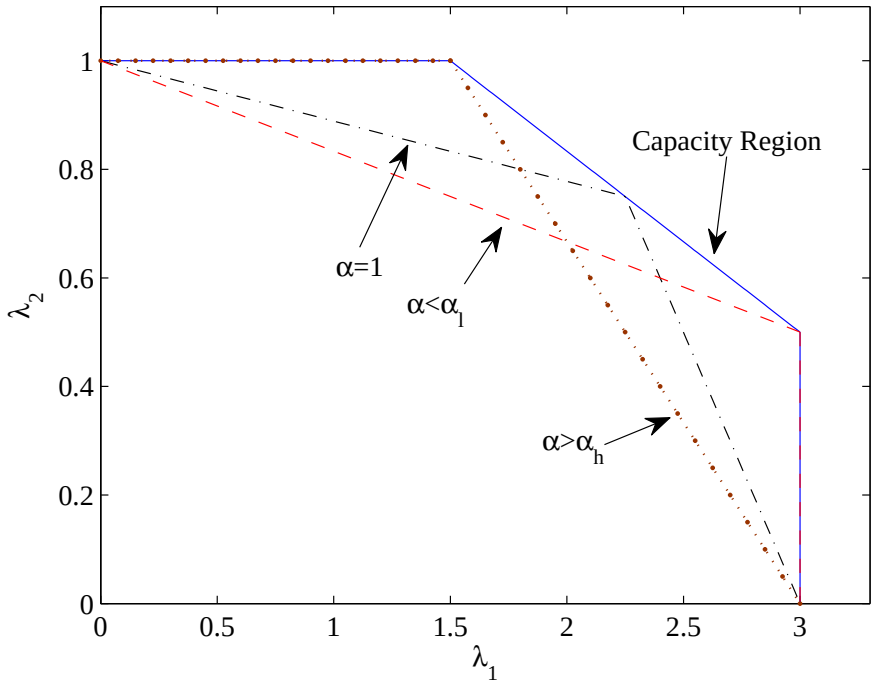


Figure 4.7: Stability Region of a system with four-state channel and α-fairness CRB scheduling, $R_1^{\mathrm{ON}} = 6$, $R_2^{\mathrm{ON}} = 2$, $\alpha_l = 0.61$, $\alpha_h = 2.71$.

and may exhibit some undesirable properties, such as decreasing the arrival rate of one flow may lead the system to be unstable, and proportionally decreasing the arrival rates of flows may lead the system to be unstable. Such properties suggest that in a system using the UB scheduling policy, reducing the traffic intensity may have a negative impact on the QoS for all on-going traffic, which is contradict to the intuition. Different from the UB scheduling policy, the stability region of the CRB scheduling policy is derived in closed-form, and is a convex hull. In addition to the stability region, we have further discussed the extended stability region. The results show that by assigning a proper weight to each user, the weighted scheduling policy can stabilize the system if the arrival process is stationary and the average arrival rate lies inside the capacity region. Simulation and numerical examples have been given to explain the analytical results and validate our analysis.

Although the CRB scheduling policy is better than the UB scheduling policy in terms of the stability region, it needs explicit knowledge of the number of users in the system, which may bring some difficulties to implement, since how frequently to update this information may be hard to design.

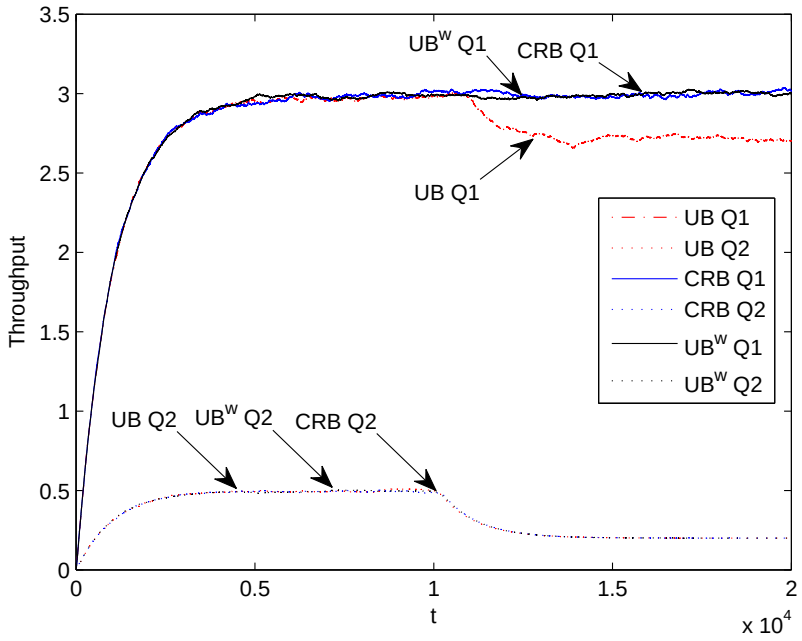


Figure 4.8: Throughput Comparison, $R_1^{\mathrm{ON}} = 6$, $R_2^{\mathrm{ON}} = 2$, $\alpha = 0.5$.

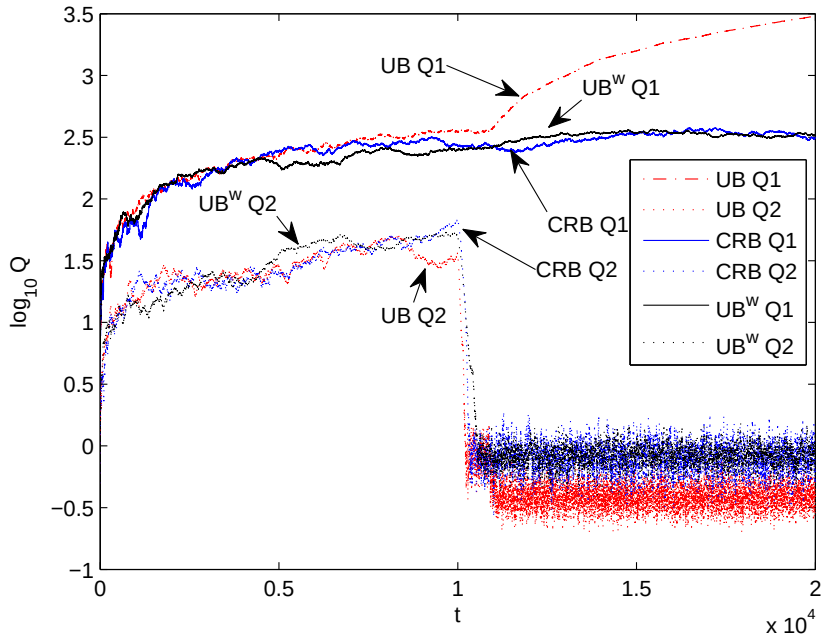


Figure 4.9: Queue Length Comparison, $R_1^{\mathrm{ON}} = 6$, $R_2^{\mathrm{ON}} = 2$, $\alpha = 0.5$.

Chapter 5

Limiting Properties of Overloaded Multiuser Wireless Systems with Throughput-Optimal Scheduling

In the previous chapters, the scheduling algorithms we discussed are in the class of algorithms that originally designed based on the assumption that the system is saturated. However, in practice, the system is not always saturated. If taking the traffic arrived in the system into account, we have shown that the stability regions of these utility-based scheduling algorithms are generally less than the capacity region. There is one type of algorithms whose stability region is equal to the capacity region. This type of algorithms is usually called throughput-optimal scheduling. In this chapter, we discuss the limiting properties of the throughput-optimal scheduling algorithms in a overloaded scenario which is generally missed in the literature.

5.1 Introduction

Throughput-optimal scheduling [68] is a class of important scheduling polices in multiuser wireless systems. It not only can explore the link quality variation as those utility-based scheduling policies discussed in [96, 77], but also can provide the maximal stability region in a network with stochastic traffic, which is superior to the utility-based scheduling whose stability region is smaller than the capacity region in general [91].

The first throughput-optimal scheduling algorithm was proposed two decades ago

[80]. The authors studied the link scheduling problem in a centralized wireless system with ON-OFF channel and proposed a MaxWeight scheduling algorithm to stabilize the system. Later, different kinds of throughput-optimal scheduling algorithms were proposed to provide diverse features under various system assumptions. The probability of max-queue overflow is asymptotically minimized by the scheduling algorithms proposed in [73, 78], [84]. The queueing delay is minimized by the algorithms proposed in [68, 69], [102, 59]. Scheduling algorithms that can provide better delay performance compared with MaxWeight were proposed in [71, 106]. The system in the presence of heavy-tailed traffic can be stabilized by the algorithms proposed in [30, 55]. General guidelines about the necessary and sufficient conditions of the throughput-optimal scheduling were discussed in [3, 12], [105, 57].

The performance of such throughput-optimal scheduling has been extensively investigated under the assumption that the system is stable, or underloaded. However, it is inevitable that a system may experience overloaded periods in practice due to the fluctuation of the traffic volume [6]. Therefore, it is important to characterize the system behavior in overloaded periods which has not received sufficient attention yet. The state-of-the-art research in this area has concluded only for some special throughput-optimal scheduling policies, such as MaxWeight scheduling in [9] and general-MaxWeight scheduling in [72]. The general system behavior of an overloaded system is still missing.

To fit the gap, in this work, we have studied the limiting properties of overloaded multiuser wireless systems with infinite buffer and a throughput-optimal scheduling policy similar to [12]. We have quantified the network performance of two throughput-optimal scheduling algorithms, the generalized MaxWeight (GMW) [12, 57] and the Log-Rule scheduling [68]. With the same throughput-optimal scheduling policy, we have further analyzed the performance of a finite-buffer system with Drop-Tail queue [14] and various buffer-sharing schemes, which is of practical interests and often missed in the literature.

We have made the following key observations: first, with infinite buffer, when the system is overloaded, all the queues in the system are unstable and the network converges to a fixed point formed by the average throughput and the scheduling function of queue length. Furthermore, the average throughput can be easily obtained by solving a convex optimization problem. Second, with the GMW algorithm, strict priority can be given to users by a proper parameter, so the QoS of users can be tuned easily. If all the users have the same priority, the system fairness in terms of Jain's

index [31] cannot be guaranteed, but the scheduler can achieve certain fairness for the blocked/queued traffic which echoes the results in [9]. With the Log-Rule scheduling, the average throughput is identical to that with an asymptotic GMW scheduling. Third, if the system is subject to a shared buffer constraint, *i.e.*, the buffer is shared among queues, then the average throughput converges to a value which is not related to the type of the throughput-optimal scheduling algorithm. If each queue has its dedicated buffer, some users might suffer starvation, and some might achieve rate stability (average throughput equals its average arrival rate), depending on their buffer sizes.

5.2 Related Work

There are two categories of work discussing the system behavior in an overloaded network. One is to design a scheduling policy towards some specific objectives, such as providing a desired throughput in [81], aiming to stabilize part of the queueing system in [24], and dynamic routing to balance the overloaded traffic in [16]; the other is to discuss the system behavior under certain scheduling policies. In [11], the system behavior of an overloaded network with α-fair scheduling was analyzed, and the asymptotic growth rate was obtained, which is a fixed-point of the system. In [72], the network behaviors of general-MaxWeight and α-fair scheduling policies in a multi-hop switched network were discussed. It showed that the queue size grows linearly with time for both scheduling policies, and the corresponding growth rates were characterized. In [9], the authors characterized the queue-size growth rate of the MaxWeight scheduling in parallel queues, and showed that the weight parameter can be tuned to achieve a certain fairness which is defined as a function of the growth rate.

Our work is different from the existing work in several aspects. First, different from [11] and [72], where a multi-hop network with fixed link service rate was discussed, we discuss a multiuser wireless system with time-varying service rate, which is similar to the network setting in [9]. Second, the throughput-optimal scheduling discussed in our work is more general than that in [72] and [9], where (general)-MaxWeight scheduling policy only was discussed. Third, we focus on the average throughput of the overloaded system, while the previous work focused on the queue size of the overloaded system. Fourth, we not only obtain the results for the infinite buffer case which is a similar assumption to the previous work, but also consider the finite buffer

case, which is missed in the literature.

5.3 System Models and Preliminaries

5.3.1 N-User Fading Broadcast Channel

We consider a system where N users communicate with a base station through a block fading broadcast channel. Within each time slot, the channel for each user is an additive white Gaussian noise (AWGN) one with a constant channel gain. Across time slots, the channel gain for each user is independent and identically distributed. The user set is denoted by $\mathcal{N}$. For users in $\mathcal{N}$, the fading processes are independent of each other, jointly stationary and ergodic, but the statistical properties are not necessarily the same. We further assume that the base station can obtain the channel state information (CSI) at the beginning of each time slot.

In time slot t, the achievable rate region is denoted by $\mathcal{C}(t)$, which is determined by the MAC layer and the physical layer protocols jointly. We further assume that time-sharing is always possible among users within each time slot (multiple users can be scheduled in one slot). Therefore no matter what kind of MAC and PHY layer technology is used, $\mathcal{C}(t)$ is always a convex and coordinate convex region, and no larger than the capacity region of the corresponding channel. For instance, if time division multiple access is used, $\mathcal{C}(t)$ is a convex and coordinate convex simplex; if superposition coding and successive interference cancellation is used, $\mathcal{C}(t)$ equals to the capacity region of an N-user degraded Gaussian broadcast channel. In addition, we assume that the maximal transmission rate for any user in time slot t is always positive, and therefore $\mathcal{C}(t)$ is always an N-dimensional region. By taking an average over time, we can obtain the average achievable rate region, which is the weighted Minkowski sum of $\mathcal{C}(t)$, denoted by $\mathcal{C}$, and is convex and coordinate convex.

5.3.2 Queueing Model

The network under consideration is a collection of queues, and each queue has a FIFO queue discipline. Data packets arrive randomly and are queued up in a buffer reserved for each user, and the arrival processes for different users are independent with each other. The resource is allocated at the beginning of each time slot based on the scheduling algorithm. Here, we first assume that the buffer size of each queue

is infinity, and the finite buffer case will be discussed in Sec.5.6. The state of the i-th buffer is the queue length and denoted by $q_i(t)$. Assume that the amount of allocated data to user i in time slot t is $r_i(t)$, whose vector form is $\mathbf{r}(t)$ and satisfies $\mathbf{r}(t) \geq \mathbf{0}$; the amount of arrived data in user i in time slot t is $a_i(t)$, whose vector form is $\mathbf{a}(t)$ and satisfies $\mathbf{a}(t) \geq \mathbf{0}$. All queue states form a vector $\mathbf{q}(t) \geq \mathbf{0}$, which is updated by: $\mathbf{q}(t+1) = [\mathbf{q}(t) - \mathbf{r}(t) + \mathbf{a}(t)]^+$, where $[\mathbf{x}]_i^+ = \max\{0, x_i\}$, $\forall i \in \mathcal{N}$. We further assume that $\{a_i(t), t = 1, 2, ...\}$ is a sequence of independent and identically distributed random variables, and $a_i(1)$ has finite moments and satisfies $\lim_{A\to\infty} \sum_i^N A f_i(A) Pr\{a_i(1) > A\} = 0$ where $f_i(.)$ is the scheduling function to be explained later. This condition is used to guarantee that the tail of the arrival distribution decays fast enough compared to the scheduling function. The average arrival rate is defined as $\boldsymbol{\lambda} = \mathbb{E}[\mathbf{a}(1)]$, where $\mathbb{E}$ is to take expectation.

5.3.3 Scheduling Policy

In this paper, we mainly focus on the throughput-optimal scheduling policy proposed in [12], as it is one of the most general scheduling policy. In order to simplify our analysis, we slightly modify the original scheduling policy, and show it as follows.

The rate allocated to users in time slot t is based on the solution to the following weighted-sum-rate-maximization problem:

$$\mathbf{r}(t) \in \arg\max_{\boldsymbol{\eta}\in\mathcal{C}(t)} \sum_i f_i(q_i(t))\eta_i, \tag{5.1}$$

and the ties are broken randomly, where η_i is the possible transmission rate of user i in slot t, and $\boldsymbol{\eta}$ is the possible transmission rate vector lies inside the instantaneous achievable rate region $\mathcal{C}(t)$, $f_i(x)$ is the scheduling function with $x \geq 0$ and satisfies the following conditions:

1) $f_i(x)$ is a non-negative strictly increasing continuous function with $\lim_{x\to\infty} f_i(x) = \infty$.

2) Given any $C_1, C_2 > 0$ and $0 < \sigma < 1$, there is some $M > 0$ such that for all $x > M$, we have $(1-\sigma)f_i(x) \leq f_i(x - C_1) \leq f_i(x + C_2) \leq (1+\sigma)f_i(x)$.

Let $\bar{\mathbf{f}}(\mathbf{x})$ be the normalized weight in vector form, whose i-th component is denoted by $\bar{f}_i(\mathbf{x}) = \frac{f_i(x_i)}{\sum_i f_i(x_i)}$. Equivalently, the optimization problem (5.1) can be represented as follows:

$$\mathbf{r}(t) \in \arg\max_{\boldsymbol{\eta}\in\mathcal{C}(t)} \sum_i \bar{f}_i(\mathbf{q}(t))\eta_i. \tag{5.2}$$

5.3.4 Stability

We adopt the definitions of stability presented in [46], which are shown as follows.

Definition 5.1. A queue q is weakly stable if, for every $\epsilon > 0$, there exists $B > 0$ such that $\limsup_{t\to\infty} \Pr\{q(t) > B\} < \epsilon$, where $q(t)$ is the queue length in time t.

Definition 5.2. A system of queues $\mathbf{q}$ is weakly stable if, for every $\epsilon > 0$, there exists $B > 0$ such that $\limsup_{t\to\infty} \Pr\{\|\mathbf{q}(t)\| > B\} < \epsilon$, where $\|\mathbf{q}(t)\|$ is the Euclidean norm of $\mathbf{q}(t)$.

From the definition we can conclude that, if q is unstable, then for any $B > 0$, we have $\limsup_{t\to\infty} \Pr\{q(t) < B\} < \epsilon$, where $\epsilon > 0$ and is arbitrarily small; if q is stable, we can find a B such that for all the t, $\Pr\{q(t) < B\} > 1 - \epsilon$, where $\epsilon > 0$ and is arbitrarily small. If a system of queues is unstable, we can conclude that at least one queue is unstable.

5.4 Limiting Properties

As shown in [12], with scheduling policy (5.1), if the queueing system is an aperiodic Markov chain and the mean arrival rate lies inside the achievable rate region, then the Markov chain is positive recurrent, or the system of queues is weakly stable. Whether the system is able to be strongly stable further depends on function f_i. As indicated in [46], weak stability implies that the offered load can be processed by the server, but the delay performance cannot be guaranteed. Consequently, if the system is overloaded, the system is unable to be weakly stable. Without confusion, stable means weakly stable, and unstable means unable to be weakly stable in the following.

5.4.1 Stability Property

From the definition of weak stability, it is unclear whether all the individual queues in the system are unstable, or only part of the queues in the system are unstable. We have Theorem 5.1 to answer this question.

Theorem 5.1. *Given infinite buffer, for a multiuser wireless system with throughput-optimal scheduling as (5.1), if the system is overloaded, then all the queues are unstable.*

Proof. Suppose that queue 1 is stable, and queue 2 is unstable. Then, we can find a B_1 such that for all t, $\Pr\{f_1(q_1(t)) < B_1\} > 1-\epsilon_1$, where $\epsilon_1 > 0$. Because queue 2 is unstable, then for any $B > 0$, we have $\limsup_{t\to\infty} \Pr\{q_2(t) < B\} < \epsilon_2$, where $\epsilon_2 > 0$. So we have $\liminf_{t\to\infty} \Pr\{q_2(t) > B\} > 1-\epsilon_2$. Because f_2 is a strictly increasing continuous function, we have $\liminf_{t\to\infty} \Pr\{f_2(q(t)) > f_2(B)\} > 1-\epsilon_2$. By choosing $B_2 = B$, we have $\liminf_{t\to\infty} \Pr\{f_2(q_2(t)) > B_2\} > 1-\epsilon_2$.

Suppose that the optimal solution for the following problem is $\boldsymbol{\eta}^*(t)$,

$$\max_{\boldsymbol{\eta}\in\mathcal{C}(t)} \sum_i w_i\eta_i,$$

where for all the i, $w_i > 0$. Because $\mathcal{C}(t)$ is an N-dimensional region, *i.e.*, we can always increase the rate of one user by decreasing the rates of other users. Therefore, by increasing w_1 and decreasing w_2, $\eta_1^*(t)$ will increase and $\eta_2^*(t)$ will decrease.

So we have, when $t \to \infty$, with probability $(1-\epsilon_1)(1-\epsilon_2)$, the rate allocated to user 1 is upper-bounded by $r_1^*(t)$, and $\mathbf{r}^*(t)$ is the solution to the following problem,

$$\max_{\boldsymbol{\eta}\in\mathcal{C}(t)} B_1\eta_1 + B_2\eta_2 + f_3(q_3(t))\eta_3 + ... + f_N(q_N(t))\eta_N. \tag{5.3}$$

Since for any B_2 and ϵ_2, we have $\lim_{t\to\infty} \Pr\{f_2(q_2(t)) > B_2\} > 1-\epsilon_2$. Consequently, for any given ϵ_1 and the corresponding B_1, with any given ϵ_2, we can choose a B_2 such that $B_2 >> B_1$. Then based on (5.3), we have $r_1^*(t) \to 0$ as $B_2 \to \infty$ and $B_2 >> B_1$.

We conclude that with probability $(1-\epsilon_1)(1-\epsilon_2)$, when $t \to \infty$, the average rate allocated to user 1 is upper-bounded by $\lim_{T\to\infty} \frac{1}{T}\sum_{\tau=t}^{t+T-1} r_1^*(\tau)$, where $\lim_{t\to\infty} r_1^*(t) \to 0$.

Since the average rate allocated to user 1 is always upper-bounded by the average achievable rate which is a finite value, with probability $1-(1-\epsilon_1)(1-\epsilon_2)$, the average rate allocated to user 1 is upper-bounded by a finite value.

In summary, we can conclude that the average rate allocated to user 1 is upper-bounded by a value which approaches to 0 as the increment of time. As a result, queue 1 is not possible to be stable, which contradicts the assumption. Thus we proved Theorem 5.1. □

As all the queues are unstable, then based on the stability properties shown in Sec.5.3.4, when $t \to \infty$, $\mathbf{q}(t) - \mathbf{r}(t) + \mathbf{a}(t) > \mathbf{0}$ holds with probability $1-\epsilon$, where ϵ is arbitrarily small, which suggests that the allocated rate will not be wasted due to the

shortage of data in the queue. Therefore, when obtaining the average throughput, we can use the allocated rate instead of the transmitted data size in each slot.

5.4.2 Average Throughput and Fixed Point of the System.

Theorem 5.2. *For an overloaded multiuser wireless system with scheduling and resource allocation algorithm as in (5.1), the corresponding average throughput of users in the system converges, i.e., for any* t_0, $\lim_{T\to\infty}\frac{1}{T}\sum_{t=0}^{T-1}\mathbf{r}(t_0+t) \to \boldsymbol{\mu}$. $\boldsymbol{\mu}$ *is a solution to the following problem*

$$\max_{\bar{\boldsymbol{\eta}}\in\mathcal{C}^{\boldsymbol{\lambda}}}\sum_i \lim_{t\to\infty}\bar{f}_i((\boldsymbol{\lambda}-\boldsymbol{\mu})t+\boldsymbol{\mu}^{'}(t))\bar{\eta}_i, \tag{5.4}$$

where $\bar{\eta}_i$ *is the possible average throughput of user* i*, and* $\bar{\boldsymbol{\eta}}$ *is the possible average throughput vector,* $\mathcal{C}^{\boldsymbol{\lambda}}=\{\bar{\boldsymbol{\eta}}|\bar{\boldsymbol{\eta}}\in\mathcal{C},\bar{\boldsymbol{\eta}}\le\boldsymbol{\lambda}\}$ *is the constrained average achievable rate region, as the average throughput should be no larger than the average arrival rate, and* $\boldsymbol{\mu}^{'}(t)$ *is an auxiliary variable with* $\lim_{t\to\infty}\boldsymbol{\mu}^{'}(t)/t=\mathbf{0}$*. Furthermore,* $\boldsymbol{\mu}$ *is unique, and is the solution to the following problem*

$$\lim_{t\to\infty}\max_{\bar{\boldsymbol{\eta}}\in\mathcal{C}^{\boldsymbol{\lambda}}}\ \sum_i F_i((\lambda_i-\bar{\eta}_i)t), \tag{5.5}$$

where $F_i(x)$ *is an antiderivative of* $f_i(x)$*.*

Proof. First we prove that the average throughput of the system converges.

For any $n>0$ we can always find $C_{1,i}$ and $C_{2,i}>0$ such that $q_i(t)-C_{2,i}\le q_i(t+n)\le q_i(t)+C_{1,i}$. Since f_i is a non-negative increasing continuous function, we have $f_i(q_i(t)-C_{2,i})\le f_i(q_i(t+n))\le f_i(q_i(t)+C_{1,i})$.

Since q_i is unstable, with probability $1-\epsilon$, for any $M>0$, there exists a T such that for all $t>T$, we have $q_i(t)>M$. Further based on condition 2) of the scheduling algorithm, for any i and $0<\sigma_i<1$, for all $t>T$, we have $(1-\sigma_i)f_i(q_i(t))\le f_i(q_i(t+n))\le(1+\sigma_i)f_i(q_i(t))$.

Then we have

$$\frac{(1-\sigma_i)f_i(q_i(t))}{\sum_i(1+\sigma_i)f_i(q_i(t))}\le\bar{f}_i(\mathbf{q}(t+n))\le\frac{(1+\sigma_i)f_i(q_i(t))}{\sum_i(1-\sigma_i)f_i(q_i(t))}.$$

Define $\Delta_i(t,n) \triangleq \bar{f}_i(\mathbf{q}(t+n)) - \bar{f}_i(\mathbf{q}(t))$, and choose $\sigma_i = \sigma$, so we have

$$(\frac{1-\sigma}{1+\sigma} - 1)\bar{f}_i(\mathbf{q}(t)) \leq \Delta_i(t,n) \leq (\frac{1+\sigma}{1-\sigma} - 1)\bar{f}_i(\mathbf{q}(t)).$$

Equivalently, we have

$$|\Delta_i(t,n)| \leq \max(\frac{2\sigma}{1+\sigma}, \frac{2\sigma}{1-\sigma})\bar{f}_i(\mathbf{q}(t)) = \frac{2\sigma}{1-\sigma}\bar{f}_i(\mathbf{q}(t)).$$

Define $\delta_i \triangleq \frac{2\sigma}{1-\sigma}$, since $\bar{f}_i(\mathbf{q}(t)) \leq 1$, we have $|\Delta_i(t,n)| \leq \delta_i$.

So for any δ_i and n we can find a corresponding σ that satisfies $|\Delta_i(t,n)| < \delta_i$.

In summary, with probability $1-\epsilon$, for any i and δ_i, we can find a T such that for all $t > T$ and any $n > 0$,

$$|\bar{f}_i(\mathbf{q}(t+n)) - \bar{f}_i(\mathbf{q}(t))| \leq \delta_i,$$

and then we can conclude that for any i, $\bar{f}_i(\mathbf{q}(t))$ is a Cauchy sequence indexed by t, thus Cauchy converges in probability.

Suppose that $\bar{f}(\mathbf{q}(t))$ converges to $\mathbf{w}$. According to (5.2), $\mathbf{r}(t)$ converges to a solution to the following problem

$$\max_{\boldsymbol{\eta} \in \mathcal{C}(t)} \sum_i w_i \eta_i,$$

which is only related to the capacity region in time slot t ($\mathcal{C}(t)$) and a weight vector ($\mathbf{w}$). Consequently, the convergence of the average throughput,

$$\lim_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T} \mathbf{r}(t_0 + t),$$

only requires the existence of the average capacity region, which is guaranteed by the assumption that the fading channel process is ergodic. Thus, the average throughput converges in probability, and we use $\boldsymbol{\mu}$ to denote it.

Then we prove that $\boldsymbol{\mu}$ is the solution to an optimization problem.

Since all the queues are unstable, with probability $1-\epsilon$, for any $M > 0$, there exists a T such that for all $t > T$, we have $\min_i q_i(t) > M$. So for all $t > T$,

$$\mathbf{q}(t) = \mathbf{q}(t-1) - \mathbf{r}(t-1) + \mathbf{a}(t-1),$$

by taking summation from T to t, we have

$$\begin{aligned}\mathbf{q}(t) = \quad & \mathbf{q}(T) + a(T) + a(T+1) + ... + a(t-1) \\ & -(r(T) + r(T+1) + ... + r(t-1)).\end{aligned}$$

Note that

$$\begin{aligned}\boldsymbol{\lambda} &= \lim_{t\to\infty} \frac{1}{t-T}(a(T) + a(T+1) + ... + a(t-1)), \\ \boldsymbol{\mu} &= \lim_{t\to\infty} \frac{1}{t-T}(r(T) + r(T+1) + ... + r(t-1)),\end{aligned}$$

so we have

$$\mathbf{q}(t) = (\boldsymbol{\lambda} - \boldsymbol{\mu})t + \boldsymbol{\mu}^{'}(t) + \mathbf{q}(T) - (\boldsymbol{\lambda} - \boldsymbol{\mu})T,$$

where

$$\lim_{t\to\infty} \frac{\boldsymbol{\mu}^{'}(t)}{t} = \mathbf{0}.$$

As $\mathbf{q}(T) - (\boldsymbol{\lambda} - \boldsymbol{\mu})T$ is finite, hence, with probability $1 - \epsilon$, when $t > T$,

$$\begin{aligned}(1-\sigma) f_i((\lambda_i - \mu_i)t + \mu_i^{'}(t)) &\leq f_i(q_i(t)) \\ &\leq (1+\sigma) f_i((\lambda_i - \mu_i)t + \mu_i^{'}(t)),\end{aligned}$$

then based on the identical approach as to prove the convergence of $\bar{f}_i(\mathbf{q}(t))$, we can first prove the convergence of $\bar{f}_i((\boldsymbol{\lambda} - \boldsymbol{\mu})t + \boldsymbol{\mu}^{'}(t))$ and then by using the squeeze theorem to prove that $\lim_{t\to\infty} \bar{f}_i(\mathbf{q}(t)) = \lim_{t\to\infty} \bar{f}_i((\boldsymbol{\lambda} - \boldsymbol{\mu})t + \boldsymbol{\mu}^{'}(t))$ in probability. Therefore $\mathbf{r}(t)$ is the solution to the following problem,

$$\max_{\boldsymbol{\eta}\in\mathcal{C}(t)} \sum_i \lim_{t\to\infty} \bar{f}_i((\boldsymbol{\lambda} - \boldsymbol{\mu})t + \boldsymbol{\mu}^{'}(t))\eta_i.$$

Then we can conclude that the average throughput $\boldsymbol{\mu}$ converges in probability and is a solution to the following problem,

$$\max_{\bar{\boldsymbol{\eta}}\in\mathcal{C}^{\boldsymbol{\lambda}}} \sum_i \lim_{t\to\infty} \bar{f}_i((\boldsymbol{\lambda} - \boldsymbol{\mu})t + \boldsymbol{\mu}^{'}(t))\bar{\eta}_i.$$

Next, we prove $\boldsymbol{\mu}$ is unique.

The antiderivative of $f_i((\lambda_i - \mu_i)t + \mu_i^{'}(t))$ is $-tF_i((\lambda_i - \mu_i)t + \mu_i^{'}(t))$. Since $\lim_{t\to\infty} \sum_i f_i((\lambda_i - \mu_i)t + \mu_i^{'}(t))\mu_i \geq \lim_{t\to\infty} \sum_i f_i((\lambda_i - \mu_i)t + \mu_i^{'}(t))\bar{\eta}_i$, *i.e.*,

$$\lim_{t\to\infty} \sum_i f_i((\lambda_i - \mu_i)t + \mu_i^{'}(t))(\mu_i - \bar{\eta}_i) \geq 0.$$

Furthermore because $f_i(x)$ is a strictly increasing continuous function, $\boldsymbol{\mu}$ is the solution to the following problem [4]

$$\lim_{t\to\infty} \max_{\bar{\boldsymbol{\eta}}\in\mathcal{C}^{\boldsymbol{\lambda}}} \ \sum_i F_i((\lambda_i - \bar{\eta}_i)t + \mu_i^{'}(t)).$$

As $\lim_{t\to\infty} \mu_i^{'}(t)/t = 0$, the above problem is further equivalent to

$$\lim_{t\to\infty} \max_{\bar{\boldsymbol{\eta}}\in\mathcal{C}^{\boldsymbol{\lambda}}} \ \sum_i F_i((\lambda_i - \bar{\eta}_i)t),$$

whose solution exists and is unique.

□

From Theorem 5.2, although the system is overloaded, the average throughput exists and converges to $\boldsymbol{\mu}$ which is a solution to (5.4). From (5.4), it is noted that $\boldsymbol{\mu}$ cannot be directly obtained. Also, by observing (5.4), we can conclude that the average throughput and the scheduling function of queue length form a unique fixed point of the system.

Since the average throughput of the system converges to the solution of (5.5), based on the above approach, we can obtain the average throughput by giving the detailed system assumptions. Note that in [9], the author obtained a similar result for the MaxWeight scheduling, which is a special case of (5.5). However, the result obtained in [9] cannot be extended to a system with a general throughput-optimal scheduling algorithm, as the result relies on the linear structure of the MaxWeight scheduling.

The scheduling algorithm according to (5.1) is an online algorithm to solve (5.5). Since $F_i((\lambda_i - \bar{\eta}_i)t)$ is a function of the average arrival rate, and $\bar{\eta}_i$ is linearly impacted by λ_i, we can conclude that the average throughput is also related to the arrival rate in general. This suggests that, in the overloaded system with scheduling algorithm as (5.1), if Jain's index (which is a function of the average throughput) is used to quantify the system fairness, then it is likely that such a fairness index is not only related to

the throughput, but also impacted by the arrival rate. This fairness issue will be elaborated in the following section, by studying two sample scheduling algorithms.

5.5 Examples: the GMW and Log-Rule Scheduling Algorithms

In this section, we study two representative throughput-optimal scheduling algorithms, the GMW [12, 57] and the Log-Rule [68]. We first discuss how to solve the optimization problem to obtain the average throughput and the impact of the parameters on the average throughput, followed by the fairness issue.

5.5.1 Generalized MaxWeight

For the GMW, we have $f_i(x) = b_i x^{\alpha_i}$, where $b_i > 0$ and $\alpha_i > 0$. Then, $F_i((\lambda_i - \bar{\eta}_i)t) = -\frac{b_i}{\alpha_i+1}(\lambda_i - \bar{\eta}_i)^{\alpha_i+1}t^{\alpha_i}$, and the average throughput is the solution to the following problem

$$\lim_{t\to\infty} \min_{\bar{\boldsymbol{\eta}}\in\mathcal{C}^{\boldsymbol{\lambda}}} \sum_i \frac{b_i}{\alpha_i+1}(\lambda_i - \bar{\eta}_i)^{\alpha_i+1}t^{\alpha_i}. \tag{5.6}$$

Easily we can see that α_i is critical to solve the problem, and the user with the largest α_i dominates the objective of (5.6). Therefore, we can adopt an iterative greedy approach to solve (5.6) as follows.

First we divide the user set $\mathcal{N}$ into K groups $\{\mathcal{G}_k\}, k = 1, 2, ..., K$. The users in the same group have the same α_i, *i.e.*, $\forall i \in \mathcal{G}_k$, $\alpha_i = \alpha_{\mathcal{G}_k}$. The groups are ordered decreasingly according to α_i. *i.e.*, if $m < n$, then $\alpha_{\mathcal{G}_m} > \alpha_{\mathcal{G}_n}$.

Suppose that the average throughput of user $i \in \bigcup_{m=1}^{k-1} \mathcal{G}_m$ is $\bar{\eta}_i^*$. Then the solution to problem (5.6) is the solution to the following problem

$$\begin{aligned} \lim_{t\to\infty} \quad \min \quad & \sum_i \frac{b_i}{\alpha_i+1}(\lambda_i - \bar{\eta}_i)^{\alpha_i+1}t^{\alpha_i-\alpha_{\mathcal{G}_k}}, \\ s.t. \qquad & \bar{\boldsymbol{\eta}} \in \mathcal{C}^{\boldsymbol{\lambda}}, \forall i \in \bigcup_{m=1}^{k-1} \mathcal{G}_m, \bar{\eta}_i = \bar{\eta}_i^*, \end{aligned}$$

which is further equivalent to

$$\lim_{t\to\infty} \quad \min \quad \sum_{i\in\bigcup_{j=1}^{k-1}\mathcal{G}_j} \frac{b_i}{\alpha_i+1}(\lambda_i-\bar{\eta}_i)^{\alpha_i+1}t^{\alpha_i-\alpha_{\mathcal{G}_k}} + \sum_{i\in\mathcal{G}_k}\frac{b_i}{\alpha_i+1}(\lambda_i-\bar{\eta}_i)^{\alpha_i+1},$$

$$s.t. \qquad \bar{\boldsymbol{\eta}}\in\mathcal{C}^{\boldsymbol{\lambda}}, \forall i\in\bigcup_{m=1}^{k-1}\mathcal{G}_m, \bar{\eta}_i=\bar{\eta}_i^*.$$

For the obtained average throughput $\bar{\eta}_i^*$, we have either $\forall i\in\bigcup_{j=1}^{k-1}\mathcal{G}_j$, $\bar{\eta}_i^*=\lambda_i$, or $\exists i\in\bigcup_{j=1}^{k-1}\mathcal{G}_j$, $\bar{\eta}_i^*\neq\lambda_i$. For the first case, the solution to (5.6) is the solution to the following problem

$$\begin{aligned} \min \quad & \textstyle\sum_{i\in\mathcal{G}_k}\frac{b_i}{\alpha_i+1}(\lambda_i-\bar{\eta}_i)^{\alpha_i+1}, && \text{(5.7a)}\\ s.t. \quad & \bar{\boldsymbol{\eta}}\in\mathcal{C}^{\boldsymbol{\lambda}}, \forall i\in\textstyle\bigcup_{m=1}^{k-1}\mathcal{G}_m, \bar{\eta}_i=\bar{\eta}_i^*. && \text{(5.7b)}\end{aligned}$$

For the second case, it implies that $\bar{\eta}_i^*<\lambda_i$, which further implies that $\bar{\boldsymbol{\eta}}^*$ lies on the boundary of $\mathcal{C}^{\boldsymbol{\lambda}}$, where $\bar{\boldsymbol{\eta}}^*=\{\bar{\eta}_i:\bar{\eta}_i=\bar{\eta}_i^*, \text{if } i\in\bigcup_{j=k+1}^{K}\mathcal{G}_j; \bar{\eta}_i=0, \text{if other wise}\}$. By summarizing the above two cases, we can conclude that the solution to (5.6) is also the solution to the following problem

$$\begin{aligned} \min \quad & \textstyle\sum_{i\in\bigcup_{j=1}^{k}\mathcal{G}_j}\frac{b_i}{\alpha_i+1}(\lambda_i-\bar{\eta}_i)^{\alpha_i+1},\\ s.t. \quad & \bar{\boldsymbol{\eta}}\in\mathcal{C}^{\boldsymbol{\lambda}}, \forall i\in\textstyle\bigcup_{m=1}^{k-1}\mathcal{G}_m, \bar{\eta}_i=\bar{\eta}_i^*,\end{aligned}$$

which further implies that the average throughputs of users in group $\mathcal{G}_k$ can be obtained by solving the above problem, as the solution is unique.

Iteratively, the average throughput is obtained.

Impact of Scheduling Parameters

By observing the algorithm structure to obtain the average throughput, we can see that the parameters α_i and b_i are important to the performance of users.

Note that in a stable system, α_i can be used to control the priority of the queue and improve the delay performance [84]. This priority only affects the delay performance, and it does not change the average throughput which equals the average arrival rate in an underloaded system. But in an overloaded system, since the scheduler allocates the available resource to the users in the decreasing order of α_i, a user with a larger α_i has a 'hard' higher priority. Therefore by increasing α_i of a user to a proper value

(for instance, larger than all the other α_j, where $j \neq i$), its average throughput can be improved. This behavior suggests that the QoS of a user can be improved by assigning a larger α_i. In summary, if the system is stable, then a larger α_i can result in a smaller delay; if the system is unstable, then it can result in a higher throughput.

Similar to α_i, b_i can also be used to differentiate the users, but within a group of users with the identical α_i. Note that the throughput of users in $\mathcal{G}_k$ is either all zero, or can be obtained from problem (5.7). As $\forall i \in \mathcal{G}_k$, α_i are identical. Fixing b_j where $j \neq i$, r_i is possible to be increased w.r.t. b_i. A larger b_i generally means a possible larger average throughput, but it cannot guarantee the user gets served first. Therefore we consider the priority associated with b_i as a 'soft' priority.

Fairness

Since α_i is used to control the priority of user, and the incoming traffic is served strictly according to the priority, the fairness should only be considered within each group. Considering the scheduling algorithm with $\alpha_i = \alpha$, the average throughput is the solution to the following problem $\min_{\bar{\boldsymbol{\eta}} \in \mathcal{C}^{\boldsymbol{\lambda}}} \sum_i b_i(\lambda_i - \bar{\eta}_i)^{\alpha+1}$, which is a $L_{\alpha+1}$-Norm minimization problem in a scaled space with constraint $\bar{\boldsymbol{\eta}} \in \mathcal{C}^{\boldsymbol{\lambda}}$, and $b_i^{1/\alpha}$ is the scale factor in dimension i.

Suppose that the scale factor for each dimension is identical, *i.e.*, $\forall i, b_i = 1$, and then geometrically, the average throughput is a point in the constraint set $\mathcal{C}^{\boldsymbol{\lambda}}$ and has the minimal $L_{\alpha+1}$ distance to the point $\boldsymbol{\lambda}$. Since $\mathcal{C}$ and $\{\bar{\boldsymbol{\eta}} | \bar{\boldsymbol{\eta}} \leq \boldsymbol{\lambda}\}$ are both convex, the constraint set $\mathcal{C}^{\boldsymbol{\lambda}}$ is also convex, so we can conclude that, as long as $\boldsymbol{\lambda}$ is not scaled proportional to $\boldsymbol{\lambda} - \boldsymbol{\mu}(\boldsymbol{\lambda})$, where $\boldsymbol{\mu}(\boldsymbol{\lambda})$ is the solution for the given $\boldsymbol{\lambda}$, the average throughput will change based on the change of $\boldsymbol{\lambda}$. As a result, the fairness (in terms of Jain's index) only makes sense for a given $\boldsymbol{\lambda}$. For a system where $\boldsymbol{\lambda}$ is not under control, fairness cannot be guaranteed as any user can change the Jain's index by increasing the arrival rate.

Although the fairness of the throughput cannot be guaranteed, the scheduling algorithm actually guarantees the fairness of the queued/blocked traffic. This can be seen from two asymptotic cases easily. When $\alpha \to 0$, the problem approaches $\max_{\bar{\boldsymbol{\eta}} \in \mathcal{C}^{\boldsymbol{\lambda}}} \sum_i \bar{\eta}_i$ which means the user with a larger possible transmission rate will be satisfied first and users with smaller possible transmission rates may starve. Such a greedy allocation can also be interpreted as that the scheduling algorithm does not consider the fairness at all. The problem approaches $\min_{\bar{\boldsymbol{\eta}} \in \mathcal{C}^{\boldsymbol{\lambda}}} \max_i |\lambda_i - \bar{\eta}_i|$ when

$\alpha \to \infty$, which means the queued traffic satisfies a min-max fairness, which is defined as the dual of the max-min fairness [65, 15]. From the above two asymptotic cases we can conclude that, the fairness of the blocked traffic can be guaranteed by choosing a proper α.

A more detailed discussion on the fairness issue in a similar system can be found in [9], where the author proposed a fairness metric using the backlog growth direction, which is identical to the fairness of the queued/blocked traffic. Furthermore, the author showed that, for a specific backlog growth direction, by designing a proper b_i, the backlogged traffic is also minimized. As [9] only discussed the fairness within each group, it is unable to find the critical impact of α_i on the fairness.

5.5.2 Log-Rule

By a slight modification to the original policy presented in [68], we have the equivalent Log-Rule which has $f_i(x) = b_i \log(1 + a_i x)$ with $a_i > 0$ and $b_i > 0$. Then

$$F_i((\lambda_i - \bar{\eta}_i)t) = \frac{b_i}{a_i t} + (\lambda_i - \bar{\eta}_i)b_i - (\frac{b_i}{a_i t} + b_i(\lambda_i - \bar{\eta}_i)) \log(1 + a_i(\lambda_i - \bar{\eta}_i)t),$$

and the average throughput is the solution to the following problem

$$\lim_{t\to\infty} \min_{\bar{\boldsymbol{\eta}} \in \mathcal{C}^{\boldsymbol{\lambda}}} \sum_i -\frac{b_i}{a_i t} - (\lambda_i - \bar{\eta}_i)b_i + (\frac{b_i}{a_i t} + b_i(\lambda_i - \bar{\eta}_i)) \log(1 + a_i(\lambda_i - \bar{\eta}_i)t),$$

which is equivalent to

$$\lim_{t\to\infty} \min_{\bar{\boldsymbol{\eta}} \in \mathcal{C}^{\boldsymbol{\lambda}}} \sum_i b_i(\lambda_i - \bar{\eta}_i) \log(1 + a_i(\lambda_i - \bar{\eta}_i)t), \tag{5.8}$$

by ignoring the terms that do not increase with t as they have no impact on the solution.

Note that the solution to (5.8) is also the solution to the following problem

$$\lim_{t\to\infty} \min_{\bar{\boldsymbol{\eta}} \in \mathcal{C}^{\boldsymbol{\lambda}}} \sum_i b_i(\lambda_i - \bar{\eta}_i) \log(1 + a_i(\lambda_i - \bar{\eta}_i)t) / \log(1 + t),$$

which is further equivalent to

$$\max_{\bar{\boldsymbol{\eta}} \in \mathcal{C}^{\boldsymbol{\lambda}}} \quad \textstyle\sum_i b_i \bar{\eta}_i, \tag{5.9}$$

since $\lim_{t\to\infty} \log(1 + a_i(\lambda_i - \bar{\eta}_i)t)/\log(1 + t) = 1$.

Problem (5.9) and problem (5.8) are not equivalent. But if (5.9) has a unique solution, then it is also the solution to (5.8).

Note that (5.9) is not related to a_i, the average throughput of the Log-Rule is only affected by parameter b_i. Comparing (5.9) to the GMW with $\alpha \to 0$ we can see that both schedulers have the same average throughput. Consequently, the discussions on the impact of b_i on the average throughput and the fairness issue are identical to the GMW with $\alpha \to 0$ case.

5.6 Performance In a Finite Buffer System

In the previous sections, we have discussed the limiting properties of the overloaded system with throughput-optimal scheduling. All the queues in the system are unstable, and each queue length increases to infinity. While with a more practical assumption that the buffer for the queue should be finite, the overloaded packets will be dropped by the queue management scheme. In this section, we give a discussion on the system performance in a finite buffer system.

5.6.1 System Assumption

Since the system is a collection of queues, all the queues can either share the same buffer, such as the downlink case of a wireless communication system, or each queue has its own dedicated buffer, such as the uplink of a wireless communication system. We further assume that the system uses the Drop-Tail scheme as the queue management scheme, and the arrival traffic for i-th flow (for user i) is a Poisson traffic with average arrival rate λ_i.

5.6.2 Shared Buffer Case

As the queue has finite buffer, the incoming packet will be dropped if it encounters the event that the buffer is full. The packet drop in a queueing system with a shared buffer is identical to that in a single queue with buffer size $B^{\max}$ and the aggregated arrival traffic.

As the incoming traffic of each flow is Poisson traffic, the aggregated traffic is also Poisson. According to the Poisson Arrival See Time Average (PASTA) property, each

packet encounters the event that the buffer is full with the same probability. So the packet dropping probability for each flow is identical and denoted as k^*. Consequently the packet dropping rate $\mathbf{d}$ is proportional to the packet arrival rate $\boldsymbol{\lambda}$, and we have $\mathbf{d} = k^*\boldsymbol{\lambda}$. For the average throughput $\boldsymbol{\mu}$, we have $\boldsymbol{\mu} = \boldsymbol{\lambda} - \mathbf{d}$. As the system is overloaded and throughput-optimal scheduling is used, we further assume the buffer size is large enough and the probability that the buffer is empty is negligible, $\boldsymbol{\mu}$ should lie on the boundary of the capacity region $\mathcal{C}$, *i.e.*, $\boldsymbol{\mu} \in \mathbf{bd}(\mathcal{C})$. Consequently, we have $\boldsymbol{\mu} = (1 - k^*)\boldsymbol{\lambda}$, and k^* is obtained from $k^* = \arg\min_{(1-k)\boldsymbol{\lambda}\in\mathcal{C}} k$.

The average throughput is not affected by the type of the throughput-optimal scheduling algorithm, and is determined by the statistical properties of the arrival traffic and the queue management scheme. Furthermore, the average throughput is proportional to the average arrival rate, which is different from the infinite buffer case where some users may starve (such as GMW with heterogeneous α_i). In other words, for an overloaded system with finite shared buffer, the long-term (permanent) fairness may be improved, despite the fairness may be poor during the transient period (as the performance during the transient period is similar to the infinite buffer case), such as in a system using the GMW scheduling with heterogeneous α_i as discussed in Sec. 5.5.1.

Note that as long as the packet dropping probability is identical for different flows, the above argument holds. Even though in a system with Drop-Tail scheme and bursty arrival traffic, the above property may not hold in general, however, certain active queue management (AQM) scheme can be used, such as Random Early Detection [14], to retain this property.

5.6.3 Dedicated Buffer Case

As discussed in [56], the system can be modeled as a controlled random walk and can be further approximated by a deterministic fluid model, where the data packets for each user are modeled as a continuous fluid flow that enter and leave the buffer [42]. Each flow has its dedicated buffer with size $B_i^{\max}$, then all the $B_i^{\max}$ will jointly determine the average throughput and therefore fairness. The corresponding fluid scheduling model is represented as $\mathbf{q}(t) = \mathbf{q}(0) - \mathbf{z}(t) + \boldsymbol{\lambda}t$, where $\mathbf{z}(t)$ is the cumulative allocated fluid resource up to time t.

As the system is overloaded and the fluid model is used to approximate the system, the system cannot be idle at the time it loses fluid. Therefore, the average throughput

should be equal to the average allocated rate, *i.e.*, $\boldsymbol{\mu} = \lim_{t\to\infty} \mathbf{z}(t)/t$. We have $\mathbf{z}(t) = \boldsymbol{\mu}t + \boldsymbol{\mu}^{'}(t)$, where $\lim_{t\to\infty} \boldsymbol{\mu}^{'}(t)/t = 0$. Consequently, we have

$$\mathbf{q}(t) = \mathbf{q}(0) - \boldsymbol{\mu}t - \boldsymbol{\mu}^{'}(t) + \boldsymbol{\lambda}t. \tag{5.10}$$

Suppose that the queues in set $\mathcal{S}$ can achieve rate stability, *i.e.*, the average throughput equals the average arrival rate, and the queues not in $\mathcal{S}$ cannot achieve rate stability. We have $\forall i \in \mathcal{S}$, $\mu_i = \lambda_i$ and $\forall i \notin \mathcal{S}$, $\mu_i < \lambda_i$. Therefore, based on (5.10), $\forall i \notin \mathcal{S}$ there exists a t_b such that for all $t > t_b$, $f_i(q_i(t)) = f_i(B_i^{\max})$, *i.e.*, the buffer of queue i is full after it has been filled up. Consequently the allocated rate in t is based on the following problem

$$\max_{\boldsymbol{\eta}\in\mathcal{C}(t)} \quad \sum_{i\notin\mathcal{S}} f_i(B_i^{\max})\eta_i + \sum_{i\in\mathcal{S}} f_i(q_i(t))\eta_i,$$

with random tie-breaking. By taking an average over time, we have $\boldsymbol{\mu}$ to be the solution to the following problem

$$\max_{\bar{\boldsymbol{\eta}}\in\mathcal{C}^{\boldsymbol{\lambda}}} \quad \sum_{i\notin\mathcal{S}} f_i(B_i^{\max})\bar{\eta}_i + \sum_{i\in\mathcal{S}} \lim_{T\to\infty} \frac{1}{T} \sum_{t=t_b+1}^{t_b+T} f_i(q_i(t))\bar{\eta}_i. \tag{5.11}$$

Since $\forall i \in \mathcal{S}$, $\mu_i = \lambda_i$ and $f_i(q_i(t)) \le f_i(B_i^{\max})$, substituting $f_i(B_i^{\max})$ for $f_i(q_i(t))$ does not change the solution. Thus, problem (5.11) is identical to the following problem

$$\max_{\bar{\boldsymbol{\eta}}\in\mathcal{C}^{\boldsymbol{\lambda}}} \quad \sum_i f_i(B_i^{\max})\bar{\eta}_i, \tag{5.12}$$

with uniform tie-breaking.

In the above analysis, the key argument is that $\forall i \notin \mathcal{S}$ and $t > t_b$, $f_i(q_i(t)) = f_i(B_i^{\max})$. Therefore, for any arrival traffic as long as the above condition can approximately hold, the average throughput is close to the solution to (5.12).

Note that if the average arrival rate $\boldsymbol{\lambda}$ is sufficiently large, the constraint set $\mathcal{C}^{\boldsymbol{\lambda}}$ will be equal to the capacity region $\mathcal{C}$. Based on (5.12), the average throughput is no longer related to $\boldsymbol{\lambda}$. Consequently, the system fairness in terms of Jain's index can be guaranteed by a properly designed buffer size $B_i^{\max}$.

By studying the system with the finite buffer assumption, we can see that the system behavior is quite different from that with infinite buffer assumption, which is typically used in the literature. The user starvation problem can be alleviated by the shared buffer scheme, while certain queues may achieve rate stability if the buffer

size is set properly in the dedicated buffer case and we will demonstrate it with an example in Sec. 5.7.1.

5.7 Performance Evaluation

In this section, we validate our analytical results and compare the system performance based on different throughput-optimal scheduling algorithms and different system assumptions. During the evaluation, Poisson arrival traffic is used if not specified. For each simulation setting, we repeat the simulation multiple runs and the results demonstrate that the average throughput can converge to the theoretical value. Then, we take one run as a sample-path of the system to plot the results in the figures. The average throughput in time slot t is calculated by taking the average over the results of the previous 2000 time slots.

5.7.1 Two-User Static Channel Case

First considering a two-user static Gaussian broadcast channel (GBC), the signal-to-noise ratio of user i is γ_i, and assume $\gamma_1 > \gamma_2$. Then the achievable rate region is [19] $\mathcal{C} = \{\mathbf{r} | r_1 = \log_2(1 + q\gamma_1), r_2 = \log_2(1 + \gamma_2) - \log_2(1 + q\gamma_2), 0 \leq q \leq 1\}$.

This channel is a special case of the general N-user fading broadcast channel, since the stochastic process governing the transition of the channel state is deterministic. Such channel is discussed because of the strictly convex property of the resultant achievable rate region, and therefore the average throughputs for different simulation settings are always unique. Also by using the two-user GBC channel first and then the Markov channel in Sec. 5.7.2, we are able to demonstrate that the convergence of $\bar{f}_i(\mathbf{q}(t))$ and average throughput does not depend on whether the channel is stochastic or not.

During the evaluation, we set $\gamma_1 = 100$ and $\gamma_2 = 10$.

Infinite Buffer Case

We first validate that with the GMW or Log-Rule scheduling, the average throughput and $\bar{\mathbf{f}}(\mathbf{q}(t))$ converge. The results are shown in Figs. 5.1 - 5.3. In the figures illustrating the average throughput, the solid curve represents the throughput of user 1 and the dashed curve represents that of user 2. The analytical results are shown by points "$\times$" in all figures.

Fig. 5.1 illustrates the system behavior with the GMW scheduler and identical α. The average throughputs and $f_1(q_1)/f_2(q_2)$ quickly converge, and different b results in different converged value. Fixing $b_1 = 1$ and increasing b_2 from 1 to 10, the average throughput of user 2 also increases, but the network does not give a 'hard' priority to user 2. From the trend of throughput changes, with the further increasing of b_2, the average throughput of user 2 will increase to the same as its arrival rate. Since $f_1(q_1)/f_2(q_2)$ is no more informative than the average throughput and the convergence of average throughput indicates the convergence of $f_1(q_1)/f_2(q_2)$, in the following we only show the comparison of the average throughput.

Fig. 5.2 shows the behavior of a system with GMW scheduler and different α. With $\boldsymbol{\alpha} = [1\ 1.3]$, the average throughput converges slowly, and cannot converge to the analytical results within 50000 time slots. Changing $\boldsymbol{\alpha}$ to [1 1.6], the system converges much fast, and the average throughput of user 2 converges to its arrival rate, which suggests that user 2 has a strictly higher priority compared with user 1.

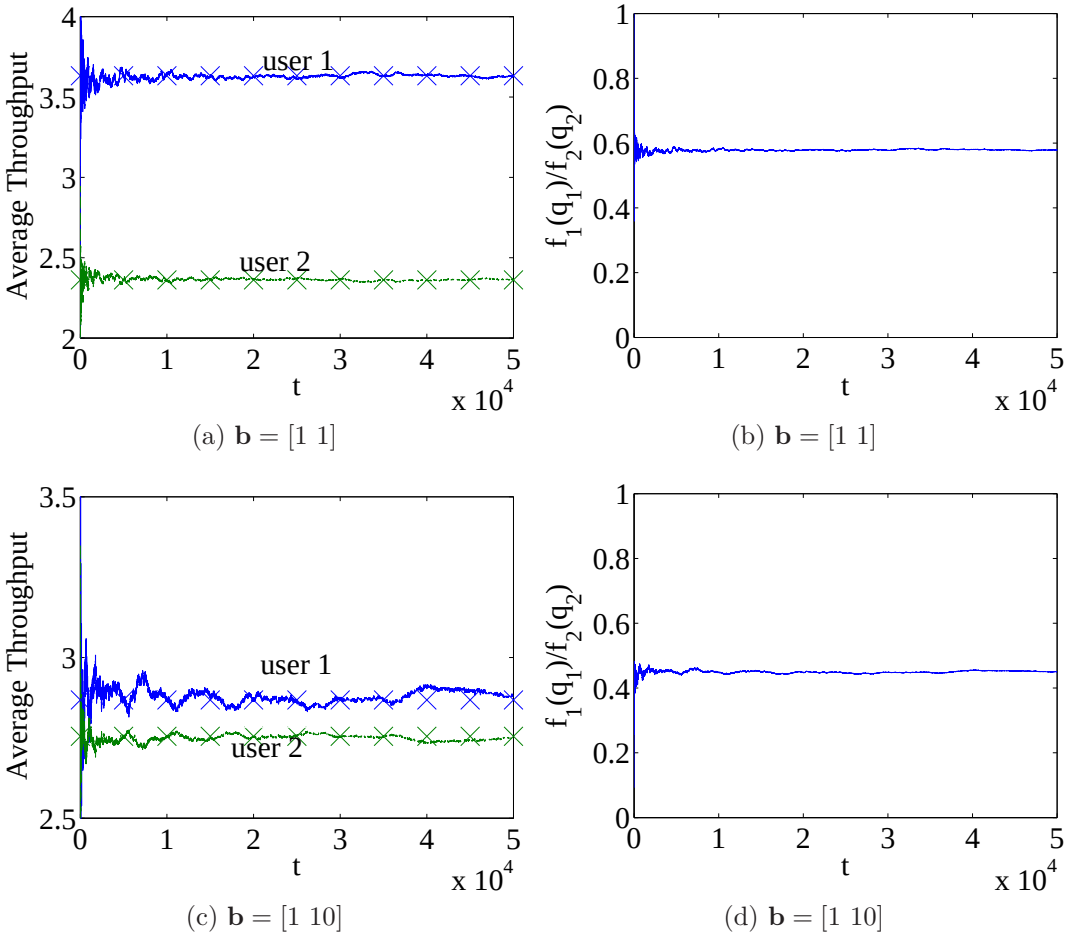
We compare the Log-Rule with the asymptotic GMW in Fig. 5.3. For the asymptotic GMW, we choose $\alpha = 0.1$. The average throughputs of GMW and Log-Rule are identical and equal to the analytical results. Also we observe that the average throughput of user 1 equals its arrival rate, which is a result that the Log-Rule or the asymptotic GMW degrades to an algorithm which schedules the user with a larger channel rate first.

Finite Buffer Case

The system behavior under the finite buffer assumption is presented here. In order to observe the transient network behavior, we set the buffer size to be a relatively large value in different scenarios.

First we show the results of the shared buffer with the Drop-Tail queue scheme case. Fig. 5.4 shows the results of a network with the GMW scheduler with different $\mathbf{b}$. As shown in the figure, the average throughput first converges to a transient value which is determined by the infinite buffer case, and thereafter converges to a permanent value. The transient value is determined by the parameters of GMW (parameter $\mathbf{b}$), while the permanent fixed value is identical and independent of these parameters.

Similar to the shared buffer with the Drop-Tail queue scheme case, in the dedicated buffer with the Drop-Tail queue scheme case, the network first converges to a



(a) $\mathbf{b} = [1\ 1]$ (b) $\mathbf{b} = [1\ 1]$

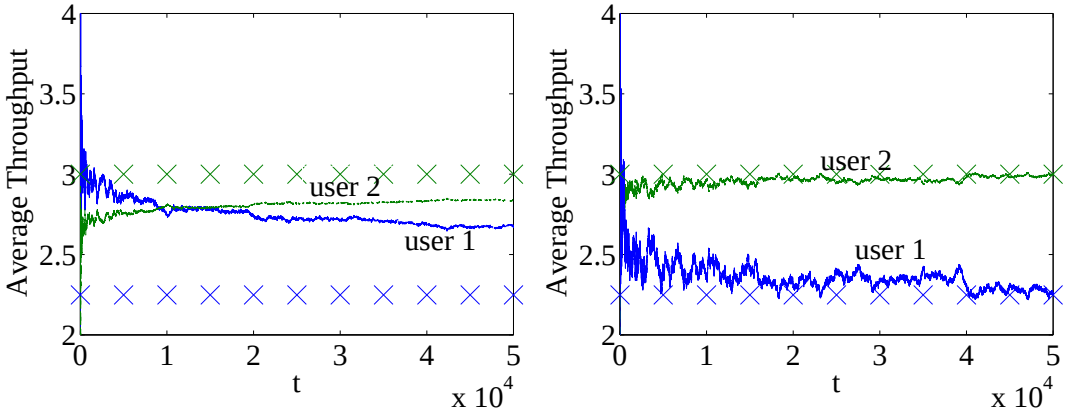
(c) $\mathbf{b} = [1\ 10]$ (d) $\mathbf{b} = [1\ 10]$

Figure 5.1: The convergence of the average throughput and $\bar{\mathbf{f}}(\mathbf{q}(t))$ of an infinite buffer network with GMW scheduler, $\boldsymbol{\alpha} = [1\ 1]$, $\boldsymbol{\lambda} = [4\ 3]$.

transient value, then converges to the permanent value, which is illustrated in Fig. 5.5. By changing the parameter $\mathbf{b}$, not only the transient value changes, but also the permanent value changes. Since the permanent throughput of user 2 equals its average arrival rate, user 2 achieves rate stability.

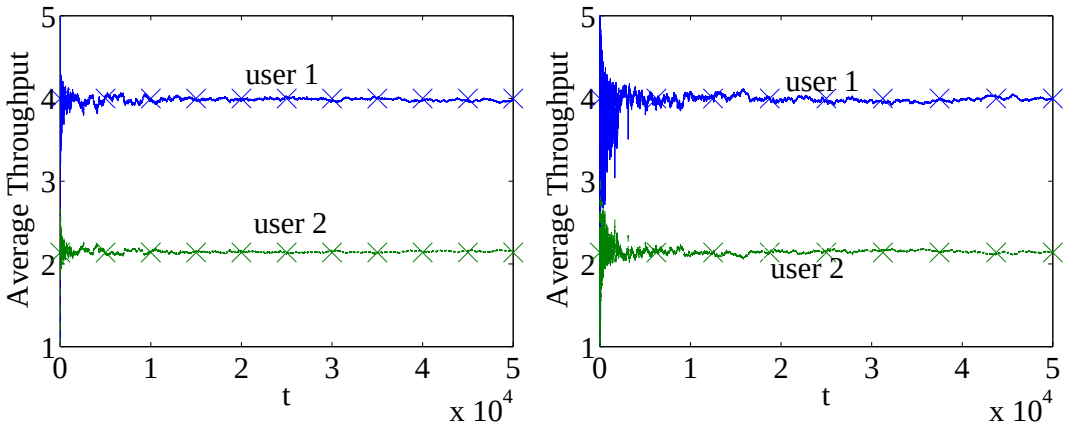
5.7.2 Markov Channel Model

We further use a Markov channel model to illustrate the system dynamics with temporarily overloaded arrival traffic. The channel of each user is independent of each other, and has two states (G and B). The transmission rate of user i in state G and B are R_i^G and R_i^B, respectively. Assume that the probability of user i in state G is π_i^G, then $\pi_i^B = 1 - \pi_i^G$. The ergodic capacity region can be obtained as



(a) $\boldsymbol{\alpha} = [1\ 1.3]$ (b) $\boldsymbol{\alpha} = [1\ 1.6]$

Figure 5.2: The average throughput of an infinite buffer network with GMW scheduler, $\mathbf{b} = [1\ 1]$, $\boldsymbol{\lambda} = [4\ 3]$.



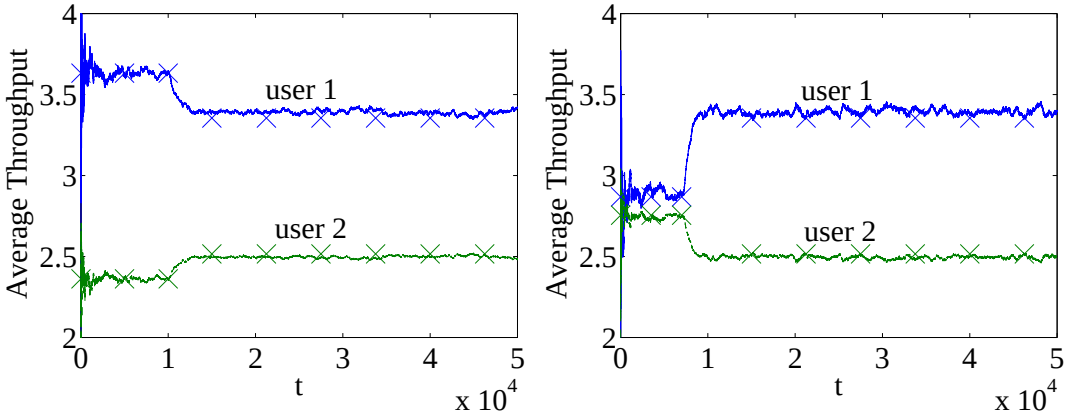
(a) Log-Rule, $\mathbf{a} = [1\ 1]$ (b) GMW, $\boldsymbol{\alpha} = [0.1\ 0.1]$

Figure 5.3: The average throughput of an infinite buffer network, comparing Log-Rule scheduler with asymptotic GMW scheduler, $\mathbf{b} = [1\ 1]$, $\boldsymbol{\lambda} = [4\ 3]$.

$\mathcal{C} = \{\mathbf{r} : r_i \leq \sum_i t_i^G \pi_i^G R_i^G + t_i^B \pi_i^B R_i^B, \sum_i t_i^B + t_i^G \leq 1\}$.

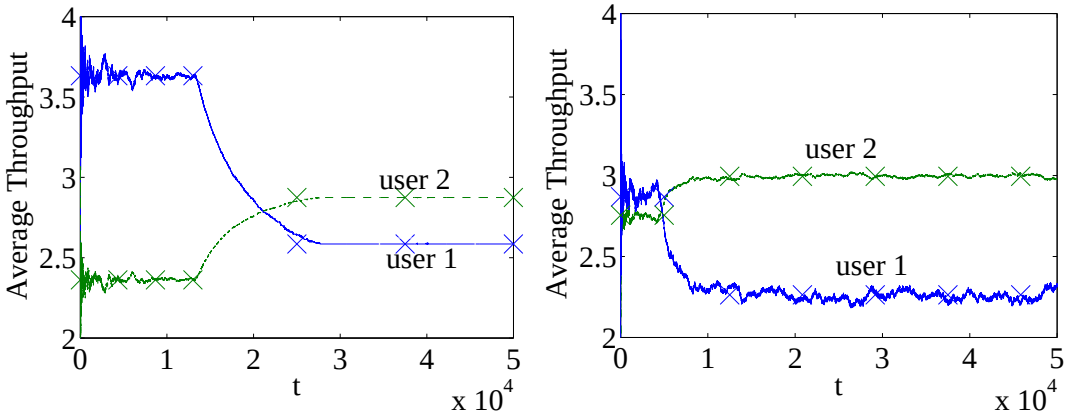
The behavior of a temporarily overloaded system is illustrated in Fig. 5.6. Here we simulate a 6-user system in 100000 time slots, and choose $\forall i, R_i^B = 1, R_i^G = 3, \pi_i^G = 1/2$. During the first 30000 time slots and the last 40000 time slots, the system has a Poisson arrival traffic with arrival rate $\boldsymbol{\lambda}$, where $\lambda_1 = \lambda_2 = \lambda_3 = 8/3$ and $\lambda_4 = \lambda_5 = \lambda_6 = 2$. From time slot 30001 to time slot 60000, there is no traffic arrived in the system.

For the shared buffer case, by investigating Fig. 5.6a, first the average throughput converges to a point determined by the infinite buffer case. After the time slot 10000,



(a) $\mathbf{b} = [1\ 1]$ (b) $\mathbf{b} = [1\ 10]$

Figure 5.4: The average throughput of a finite shared buffer network with GMW scheduler and Drop-Tail scheme, $\boldsymbol{\alpha} = [1\ 1]$, $\boldsymbol{\lambda} = [4\ 3]$, $B^{\max} = 10^4$.



(a) $\mathbf{b} = [1\ 1]$ (b) $\mathbf{b} = [1\ 10]$

Figure 5.5: The average throughput of a finite dedicated buffer network with GMW scheduler and Drop-Tail scheme, $\boldsymbol{\alpha} = [1\ 1]$, $\boldsymbol{\lambda} = [4\ 3]$, $\mathbf{B}^{\max} = [5000\ 12500]$.

as the buffer is full, the average throughput converges to the point determined by the finite buffer case. After the time slot 30000, since there is no new arrival traffic, the system can be stabilized and the average throughput converges to the same value, which is due to the symmetric channel. After the time slot 60000, a similar pattern as that during time slots 1 - 30000 can be found.

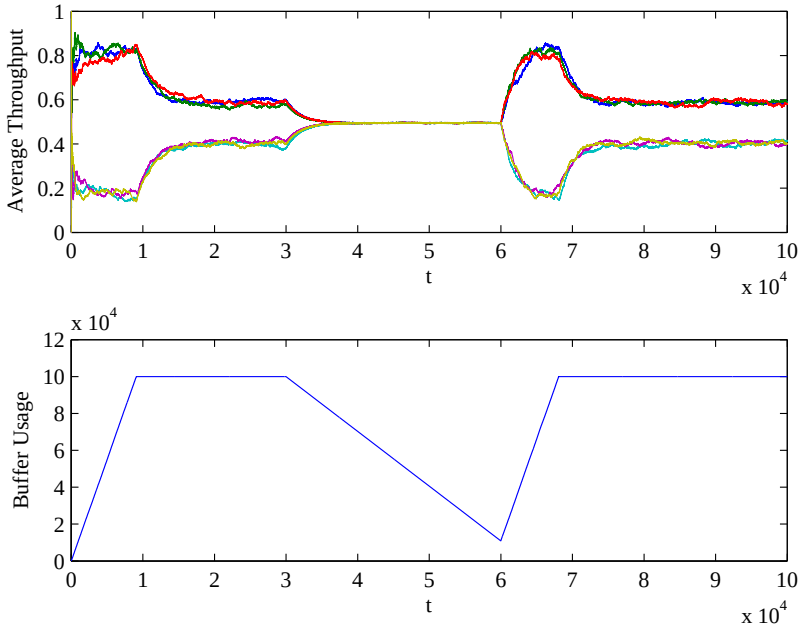
The results of the dedicated buffer case are shown in Fig. 5.6b. The curves from top to down in the buffer usage figure are for user 1 to user 6, respectively. The average throughput exhibits a similar trend as that in the shared buffer case. The buffer usages of different users in the transient period (none of the buffer is full) are

identical, which is a result of the symmetric channel assumption.

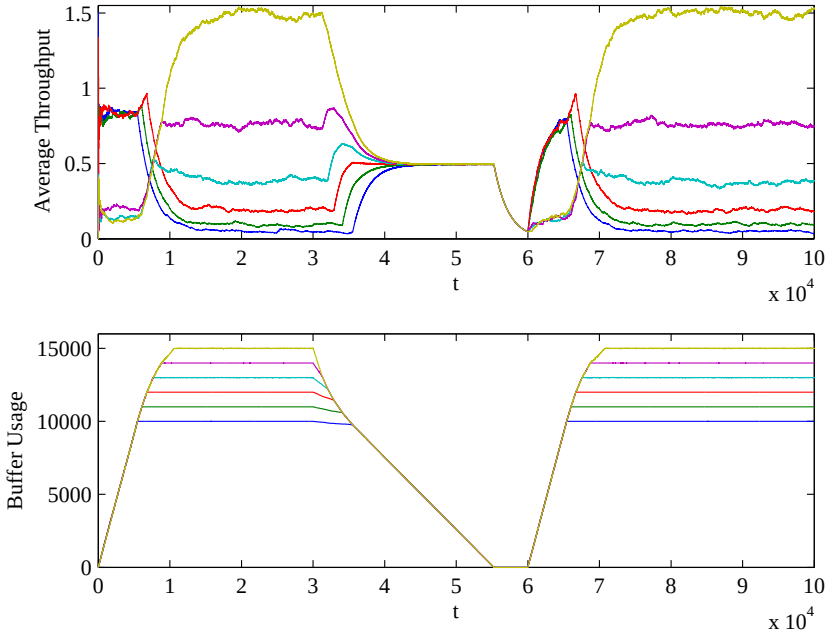
In summary, the simulation results have validated our analytical model and confirmed our analytical results and conclusions.

5.8 Conclusion and Further Discussion

In this chapter, we have studied the limiting properties of an overloaded multiuser wireless system with throughput-optimal scheduling. By studying a general throughput-optimal scheduling, we have found that certain results obtained in a system with special throughput-optimal scheduling is possibly universal. More specifically, we have shown that if the system is subject to infinite buffer assumption, all the queues in the network are unstable, but the average throughput and the scheduling function of queue length converges, respectively. By studying GMW and Log-Rule algorithms, we have found that the fairness of the corresponding system generally cannot be guaranteed. The Log-Rule can be viewed as a special GMW, and GMW can provide user differentiation by choosing parameters properly. If the buffer size is finite, then the buffer and queue management schemes play an important role on the network performance, and a proper design may alleviate the potential starvation problem.



(a) Shared Buffer, $B^{\max} = 10^5$.



(b) Dedicated Buffer, $B_i^{\max} = 10000 + (i-1) \times 1000, i = \{1, 2, 3, 4, 5, 6\}$.

Figure 5.6: The system behavior of a finite buffer network with GMW scheduler and Drop-Tail scheme. $\boldsymbol{\alpha} = \mathbf{1}$, $\mathbf{b} = \mathbf{1}$.

Chapter 6

Secrecy Outage Probability in Multiuser Wireless Systems with Stochastic Traffic

In Chapter 2 and Chapter 3, we have discussed how to design scheduling algorithms that improve the system performance by using advanced physical layer technologies. Due to the broadcast nature of the wireless channel, the security is also an important aspect in multi-user wireless systems that needs to be considered. In this chapter, we study the scheduling problem in a system using the information-theoretic security encoder/decoder, which is one of the key physical layer security technologies. We first extend the definition of the secrecy outage probability to wireless systems with adaptive transmission rates. The scheduling problem in the aforementioned system, jointly considering the reliability, security and stability, is studied. Stochastic network optimization framework is used to decompose the problem and an online algorithm ONE is proposed. We further consider an offline alternative problem, discuss the optimal solution and show that the aforementioned algorithm ONE cannot lead to optimal solution in some scenarios. By comparing the offline algorithm with algorithm WSSTRM, we have proposed a refined online algorithm WSSTRM-R which is an optimal algorithm. Extensive simulations are conducted to show the impact of the information arrival rate and the channel conditions on the system secrecy outage probability. These observations provide important insights and guidelines for the design and resource management of future wireless networks using secure communication technologies.

6.1 Introduction

In a wireless system, there are several aspects that affect the system performance, such as capacity, reliability and security. Traditionally, security is a high-layer issue, and is designed independently of the network protocol. But this approach may have some drawbacks. For instance, an application-layer solution may require a higher computational complexity that may not be desirable for energy-limited devices such as smart phones. Recently, physical-layer security became an attractive research area, since it can provide different kinds of security solutions in wireless systems, by exploring the physical-layer features such as channel conditions that are traditionally overlooked.

Physical-layer security in wireless systems has been widely discussed from different aspects [76]. For instance, due to the unique randomness of the channel, the channel information can be used to generate a secret key in a wireless network, which is discussed in [20, 87, 51]. The uniqueness feature can also be used as the link signature for authentication as discussed in [63, 103, 53]. The spread spectrum communication has been revisited as a physical-layer security approach in [48, 29]. Cooperative jamming and artificial noise are used to improve the secrecy capacity region as discussed in [18, 75].

Although these designed security schemes utilize the uniqueness of the physical-layer information, most of them are designed from a traditional security viewpoint. In this chapter, we adopt a more fundamental treatment towards the security issue, *i.e.*, from the information-theoretical security viewpoint towards the confidentiality issue in multiuser wireless systems.

We study the scheduling problem in multiuser wireless systems, where one eavesdropper exists in the system. The traditional approach tries to maximize the ergodic achievable rate of the system (see, e.g., [21, 36]), which captures the fundamental capacity limits under perfect secrecy, but may exhibit a large delay due to the inherent requirement of the coding scheme for the perfect secrecy over a fading channel. Differently, we consider minimizing the secrecy outage probability of the system, which is a coding-delay-limited metric that is of practical interests. Besides, we further consider the queue stability issue which is often ignored in the work that maximizes the ergodic achievable rate. Therefore, the scheduling problem is formulated as an optimization problem minimizing the system secrecy outage probability (security issue) and subject to the constraints that the queues in the system should be stable (stability issue)

and the transmission rate does not exceed the capacity region (reliability issue).

Little work has been done jointly considering these three aspects. Some works assume that the eavesdroppers' channel state information (CSI) at symbol level (full instantaneous CSI) can be obtained by the BS, such as [50, 22, 54], which may not be practical. Some works, such as [64], relax the assumption on the instantaneous CSI, however, the designed scheme is not scalable to a case with multiple legitimate receivers, which limits the usage of the proposed algorithm. In our work, we design a scalable scheduling algorithm with a weak assumption that only the distribution of the CSI of the eavesdropper is known by the BS, which is more practical.

The contributions of this chapter are four-folds. First we have extended definition of the secrecy outage probability to wireless systems with channel-adaptive transmission. Second, we have proposed two online algorithms for the aforementioned scheduling problem, and showed that directly applying the stochastic network optimization framework cannot yield an optimal solution and some modifications should be done. Third, we have discussed an alternative offline problem, proposed an optimal offline algorithm which motivates us to design the online optimal algorithm. Fourth, we have elaborated the impact of the information arrival rate and the channel conditions on the system secrecy outage probability through extensive simulations.

The rest of this chapter is organized as follows. The preliminaries about the physical-layer security and the related work are presented in Section 6.2. System models are introduced in Section 6.3. Secrecy outage probability is revisited and the problem is formulated in Section 6.4. Online and offline algorithms are discussed in Section 6.5. A case study that the eavesdropper's channel is a non-fading additive white Gaussian noise (AWGN) channel is presented in Section 6.6, followed by the evaluation in Section 6.7. We conclude this chapter in Section 6.8.

6.2 Preliminaries and Related Work

6.2.1 Physical-Layer Security

Security is an important issue in communications, which typically include confidentiality, integrity, authentication, and nonrepudiation. The confidentiality guarantees the legitimate receivers can obtain the information, while eavesdroppers are unable to understand the information. Traditionally, confidentiality is achieved by cryptographic techniques, which are based on the computational complexity theory and

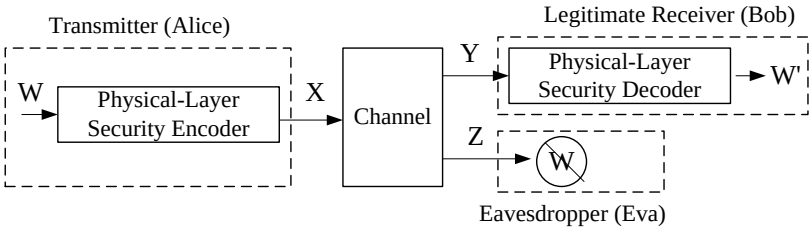


Figure 6.1: Wire-tap Channel

key distribution techniques. While for a wireless network, due to the broadcast nature of the wireless medium, the secret key distribution becomes a difficult problem [49]. The information-theoretical security, one branch of the physical-layer security, which aims to provide an alternative solution to the confidentiality, treats the secrecy communication from an information entropy point of view.

Typically, the eavesdropping in a wireless network can be captured by a wire-tap channel as shown in Fig. 6.1. The transmitter Alice has a message W intended to transmit to a legitimate receiver Bob, through a channel. The message W is mapped to the codeword X by a physical-layer security encoder, which jointly considers the security and reliability. Then X is transmitted to Bob through a wireless channel. Due to the broadcast nature of the channel, both Bob and the eavesdropper Eva can observe the corrupted messages, Y and Z. The decoder in Bob maps the received Y to an estimated message W'. The purpose of the encoder and decoder is to ensure that the estimated message is the same as the original one, i.e., $W' = W$, and the corrupted message Z received by Eva contains no information about W.

In a more practical scenario, if the channel is an AWGN channel, i.e., X is corrupted by an additive white Gaussian noise, the secrecy capacity of such a system is [45]

$$C_s = [C_Y - C_Z]^+,$$

where C_Y and C_Z are the capacity of the Bob's channel $X - Y$ and Eva's channel $X - Z$, respectively.

This result suggests that a perfect secrecy can be achieved if the entropy of the original message W is no greater than the secrecy capacity, *i.e.*, $H(W) \leq C_s$. Otherwise, part of W can be decoded from Z.

6.2.2 Related Work

Scheduling and resource allocation in a secure wireless communication system has been widely discussed in the literature. However, most of the works took a traditional information-theoretical perspective, i.e., quantifying the capacity region under different network settings. These works tried to solve an optimization problem, implicitly or explicitly, based on the assumption that the system is saturated and each user in the system always has data to transmit. For instance, the secrecy capacity region of a wire-tap channel is discussed in [98]; that of a Gaussian wire-tap channel in [45]; that of a fading channel in [21]; that of a fading broadcast channel in [36]; that of a MIMO broadcast channel in [62]. All these works only considered the reliability and security issue in communications, and ignored the stability issue which is typically treated in the higher layers. However, the stability is of equal importance with reliability and security, since it further determines whether a practical system can work properly and desirably over a sufficiently long time period.

There is little work jointly considering these three aspects. In [50], authors studied how to transmit confidential messages to users in a fast-fading broadcast wireless network, subject to three constraints: the reliability constraint that the message can be perfectly decoded, the security constraint that the message is perfectly secured and the stability constraint that the system is queue-length stable. An achievable secrecy rate region was obtained and a max-weight type of scheduling algorithm along with the optimal power control policy was designed so to satisfy these three requirements. In [22], a secure communication system was designed to achieve a constant transmission rate. In this design, the developed scheme sends the key with the data when the system is perfectly secured, and uses the key to protect the data when the system is subject to a secrecy outage. A power control scheme has also been designed to maximize the transmission rate. A work similar to [22] was reported in [54] where a different objective is used. All the above works share the same system assumption that the instantaneous CSI of the eavesdropper should be known by BS, which may not be practical.

In [64], the power allocation problem of a secure wireless communication system in the presence of statistical queueing constraints was studied. The effective secure throughput region is obtained through an effective capacity method, and a power allocation scheme that achieves such a region has been obtained. The obtained scheme implicitly considers the stability issue of the system, since a queue constraint is em-

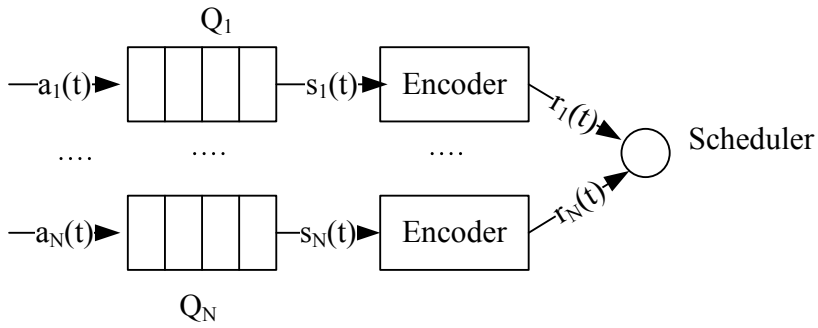


Figure 6.2: System Block Diagram

ployed. However, the authors only considered the single legitimate receiver, and the designed scheme is not scalable to a multi-legitimate-receiver case, which limits the usage of the algorithm.

6.3 System Models

We consider the downlink of a wireless network, with one base station (BS), N independent legitimate receivers and one eavesdropper. The multiple eavesdropper case can be easily extended as discussed in [94]. There are confidential data that arrive at the BS and need to be transmitted to the legitimate receivers through a shared wireless fading channel. In order to protect the data against the eavesdropper, the data have been encoded using the physical-layer security technology before transmission. The system is a time-slotted one and without loss of generality, we further assume that the slot length is 1 second. The system model is shown in Fig. 6.2.

6.3.1 Queueing Model

We assume that the data packets arrive at the end of each time slot and are queued in an infinite-size virtual buffer reserved for each legitimate user. The amount of the data arriving in time slot t for user i, $a_i(t)$, is a random variable with finite moments and cannot be transmitted until slot $t+1$. Assume that the amount of the data of user i being transmitted in the same time slot to the physical-layer security encoder is $s_i(t)$. The queue dynamic is as follows

$$Q_i(t+1) = Q_i(t) - s_i(t) + a_i(t),$$

where $Q_i(t)$ is the amount of the data buffered in queue i in time slot t, and $s_i(t) \leq Q_i(t)$ since the transmitted data size cannot be larger than the buffered data size.

6.3.2 Physical-Layer Security Encoder

The encoder uses Wyner's encoding scheme [98] to encode the input data $s_i(t)$, and the output data size is $r_i(t)$, which should be equal to the available channel resource that is allocated to user i in time slot t. The output data size should be no less than the input data size, i.e., $r_i(t) \geq s_i(t)$, and the difference $r_i(t) - s_i(t)$ quantifies the ability to secure against the eavesdropper.

6.3.3 Channel Model

The output data from the physical-layer security encoder have been directly sent through a wireless channel. For any time slot t, the received signals by legitimate receiver i, denoted by $y_i(t)$, and by the eavesdropper, denoted by $y_e(t)$, are given by, respectively

$$y_i(t) = g_i(t)x_i(t) + w_i(t),$$
$$y_e(t) = g_e(t)x_i(t) + w_e(t),$$

where $g_i(t)$ and $g_e(i)$ are the complex fading coefficients from the BS to the legitimate receiver i and the eavesdropper, respectively. $w_i(t)$ and $w_e(t)$ represent the independent and identically distributed (i.i.d.) additive Gaussian noise with unit variance at the legitimate receiver i and the eavesdropper, respectively. Therefore, the channel gains from the BS to the legitimate receiver i and the eavesdropper are $\gamma_i(t) = |g_i(t)|^2$ and $\gamma_e(t) = |g_e(t)|^2$, respectively.

Furthermore, we assume that the channel of each user is independent and each channel experiences a block fading, i.e., the channel gain remains constant during each time slot and changes independently across time slots. The fading process is assumed to be ergodic and the distribution is bounded. The duration of each time slot is long enough and Wyner's encoding scheme can be performed within each time slot.

The BS can obtain the instantaneous CSI of the legitimate receivers in each time slot, but can only know the distribution of the channel fading between the BS and the eavesdropper. As a result, $\{\gamma_e(\cdot)\}$ are i.i.d. random variables and $\{\gamma_i(\cdot)\}$ are known

by the BS.

Assume that in each time slot, only one user can transmit its data, but the user is not necessarily use all the time portion in one slot. The resource allocated to user i in time slot t used for transmission is $r_i(t)$ satisfying

$$r_i(t) \leq \tau_i(t) \log(1 + p(t)\gamma_i(t)),$$

where $p(t)$ is the allocated power in time slot t and $\tau_i(t)$ is the time portion used for transmission. Note that $\tau_i(t) \leq 1$, so the above equation guarantees the reliable communication between legitimate users and the BS. We further assume that the system is subject to a peak power constraint in each time slot, *i.e.*, $p(t) \leq 1$ and we assume that the maximal power is one.

6.4 Secrecy Outage Probability Revisited and Problem Formulation

Since the BS does not know the instantaneous CSI of the eavesdropper's channel, it is inevitable that secrecy outage happens. In this section we first revisit the secrecy outage probability defined in the literature for a single-user wireless system with a constant transmission rate, and discuss how the existing definition can be extended to a single-user wireless system with channel-adaptive transmission rates.

For the system illustrated in Fig. 6.1, in the literature, there are two distinct definitions of secrecy outage probability. In [5], the secrecy outage event is defined as $\mathcal{O}(s) := \{C_s < s\}$, where s is the target secrecy rate from Alice to Bob. The secrecy outage probability is defined as

$$P^{\text{out}} = \mathbb{P}(C_s < s). \tag{6.1}$$

As pointed out in [107], such a definition of the secrecy outage event does not distinguish between reliability and security, therefore may not be a proper design metric.

In [107], the author designed an alternative secrecy outage probability, which is a conditional probability as

$$P^{\text{out}} = \mathbb{P}(C_e > r - s|\text{message transmission}), \tag{6.2}$$

where r is the transmission rate from Alice to Bob, and C_e is the channel capacity from Alice to Eva.

The above two definitions of the secrecy outage probability only suit for the case that s is a constant during transmission. But in practice, if Alice can observe Bob's channel and obtain the channel state information, then Alice can adaptively choose s to minimize the secrecy outage probability. In the following, we extend the secrecy outage probability to such a case.

6.4.1 Block-Level Secrecy Outage Probability

First note that, the data are transmitted block-by-block in a time-slotted wireless communication system. A direct extension of the secrecy outage probability with constant transmission rate is obtaining the secrecy outage probability slot by slot and taking the average to obtain the average secrecy outage probability.

Consequently the secrecy outage probability in time slot t can be obtained as

$$P^{\text{out}}(t) = \mathbb{P}(C_e(t) > r(t) - s(t) | \text{message transmission}). \tag{6.3}$$

Since the message transmission means $s(t) > 0$, and we further have

$$C_e(t) = \tau(t)\log(1 + p(t)\gamma_e(t)), \tag{6.4}$$

$$r(t) = \tau(t)\log(1 + p(t)\gamma(t)), \tag{6.5}$$

where $p(t)$ and $\tau(t)$ is the power and time portion allocated to the user, so (6.3) can be further simplified as

$$P^{\text{out}}(t) = 1 - F(\frac{(1 + p(t)\gamma(t))\exp(-\frac{s(t)}{\tau(t)}) - 1}{p(t)}), \text{for } s(t) > 0 \tag{6.6}$$

where F is the cumulative distribution function (CDF) of γ_e[1], and $P_{\text{out}}(t)$ is not defined for $s(t) = 0$.

Then, the average secrecy outage probability at block level can be obtained as

$$\bar{P}^{\text{out,BL}} = \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} P^{\text{out}}(t). \tag{6.7}$$

[1]Since $\{\gamma_e(\cdot)\}$ are i.i.d. random variables, t can be ignored.

6.4.2 Bit-Level Secrecy Outage Probability

Note that (6.7) can only quantify how frequently the message is leaked to the eavesdropper and cannot quantify the percentage of the information that is leaked to the eavesdropper, which can be quantified by the average secrecy outage probability at bit level as

$$\bar{P}^{\text{out,b}} = \lim_{T\to\infty} \frac{\sum_{t=1}^{T} s(t) P^{\text{out}}(t)}{\sum_{t=1}^{T} s(t)}, \tag{6.8}$$

where $P^{\text{out}}(t)$ is the same as (6.6).

Note that due to the special structure of (6.8), when obtaining $\bar{P}^{\text{out,b}}$, the defined domain of (6.6) is relaxed to $s(t) \in [0, +\infty)$.

6.4.3 Comparison

If $s(t)$ is a constant for different t, then $\bar{P}^{\text{out,b}}$ is identical to $\bar{P}^{\text{out,BL}}$, and the two secrecy outage probabilities coincide with each other. If $s(t)$ can change over time, then generally the two probabilities are different, and the block-level secrecy outage probability might not be a good metric in a system with adaptive transmission rates.

Suppose that the average target secrecy rate is $\bar{R}_s$, $\gamma(t) = \gamma$, and $p(t) = 1$. From (6.6) and (6.7), we can observe that in order to minimize $\bar{P}^{\text{out,BL}}$, $\tau(t) = 1$. Consider the following two transmission schemes. The first one is that for every slot t, we have $s(t) = \bar{R}_s$. The second one is that we choose $s(t) = \epsilon$ with probability $\frac{\log(1+\gamma)-\bar{R}_s}{\log(1+\gamma)-\epsilon}$ and choose $s(t) = \log(1+\gamma)$ with probability $\frac{\bar{R}_s-\epsilon}{\log(1+\gamma)-\epsilon}$. If (6.3) is a concave function, then the second scheme can achieve a smaller average secrecy outage probability, but only negligible data are transmitted to Bob without leaking the data to Eva. Consequently, the block-level secrecy outage probability cannot properly reflect whether the information is leaked to the eavesdropper or not in some scenarios. On the contrary, the bit-level secrecy outage probability is a direct, quantitative measure of the information leak, which is a more proper performance metric. In the following, we only discuss the bit-level secrecy outage probability, and the superscript 'b' is omitted.

6.4.4 Problem Formulation

We are interested in how to provide a best-effort security solution, since the secrecy outage may be inevitable.

In each time slot, the scheduler determines how much data ($s_i(t)$) should be fetched from the queue and sent to the encoder, and determines how to protect the data by choosing an appropriate output data size of the encoder ($r_i(t)$). Meanwhile, the system should be stabilized if possible, *i.e.*, queues in the system should be stable and the average queue length over time is bounded.

In order to achieve a high level of secrecy, we need to minimize the secrecy outage probability of the system, which is defined as the average weighted secrecy outage probability of each user, *i.e.*, $\sum_i u_i \bar{P}_i^{\text{out}}/N$, where u_i is the weight assigned to user i, and

$$\bar{P}_i^{\text{out}} = \lim_{T\to\infty} \frac{\sum_{t=1}^{T} I_i(t) s_i(t) P_i^{\text{out}}(t)}{\sum_{t=1}^{T} s_i(t)}, \tag{6.9}$$

$$P_i^{\text{out}}(t) = 1 - F\left(\frac{(1+p(t)\gamma_i(t))\exp(-\frac{s_i(t)}{\tau_i(t)}) - 1}{p(t)}\right), \tag{6.10}$$

where $I_i(t) \in \{0,1\}$ indicates which user is selected in time slot t for transmission, and satisfies $\sum_i I_i(t) \leq 1$.

Therefore, the scheduling problem can be formulated as:

$$\min \quad \sum_i u_i \bar{P}_i^{\text{out}}/N \tag{6.11a}$$

$$s.t. \quad \forall i,\ Q_i \text{ is stable}, \tag{6.11b}$$

$$\forall i,\ s_i(t) \leq \min(\tau_i(t)\log(1+p(t)\gamma_i(t)), Q_i(t)), \tag{6.11c}$$

$$\forall i, \tau_i(t) \leq 1, \tag{6.11d}$$

$$\sum_i I_i(t) \leq 1, I_i(t) \in \{0,1\} \tag{6.11e}$$

$$p(t) \leq 1. \tag{6.11f}$$

Because $\forall i$, Q_i is stable and every user achieves the rate stability, we have

$$\lim_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} s_i(t) = \lim_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} a_i(t) = \lambda_i.$$

Further, we can observe that the optimum is obtained only if $p^*(t) = 1$. This

is because the CDF function F is a monotonically increasing function, and $((1 + p(t)\gamma_i(t)) \exp(-s_i(t)/\tau_i(t)) - 1)/p(t)$ is a monotonically increasing function of $p(t)$. Therefore $\bar{P}_i^{\text{out}}$ is minimized when $p(t)$ is maximized. As a result, problem (6.11) can be reformulated as

$$\begin{aligned} \max \quad & \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \sum_i \frac{u_i I_i(t) s_i(t)}{\lambda_i N} F((1+\gamma_i(t)) e^{-\frac{s_i(t)}{\tau_i(t)}} - 1) \\ s.t. \quad & \forall i,\ Q_i \text{ is stable}, \\ & \forall i,\ s_i(t) \le \min(\tau_i(t) \log(1+\gamma_i(t)), Q_i(t)), \\ & \forall i, \tau_i(t) \le 1, \\ & \textstyle\sum_i I_i(t) \le 1, I_i(t) \in \{0,1\}, \end{aligned}$$

which is a special case of the weighted-sum secure transmission rate maximizing problem as follows

$$\max \quad \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \sum_i w_i I_i(t) R_i^s(t) \tag{6.12a}$$

$$s.t. \quad \forall i,\ Q_i \text{ is stable}, \tag{6.12b}$$

$$\forall i,\ s_i(t) \le \min(\tau_i(t) \log(1+\gamma_i(t)), Q_i(t)), \tag{6.12c}$$

$$\forall i, \tau_i(t) \le 1, \tag{6.12d}$$

$$\textstyle\sum_i I_i(t) \le 1, I_i(t) \in \{0,1\}, \tag{6.12e}$$

where $R_i^s(t) = s_i(t)(1 - P_i^{\text{out}}(t))|_{p(t)=1}$ is defined as the secure transmission rate of user i in time slot t, and w_i is the weight assigned to user i.

Note that in the above formulation we assume that the arrival rate λ_i is known to the scheduler. If λ_i is unknown, by substituting λ_i with $\frac{1}{t}\sum_{k=1}^{t} a_i(k)$, we can obtain the equivalent problem formulation.

6.5 Weighted-Sum Secure Transmission Rate Maximization

6.5.1 Online Algorithm

According to the stochastic network optimization theory [60], in order to stabilize the system, we can minimize the Quadratic-Lyapunov-drift bound. If the drift bound satisfies certain conditions, then with the drift-bound-minimizing method, the system

is stable.

Define the quadratic Lyapunov function of the system as

$$L(\mathbf{Q}(t)) = \frac{1}{2}\sum_i Q_i(t)^2,$$

then the one-slot conditional Lyapunov drift is

$$\Delta(\mathbf{Q}(t)) = \mathbb{E}[L(\mathbf{Q}(t+1)) - L(\mathbf{Q}(t))|\mathbf{Q}(t)].$$

After calculation, we have

$$\Delta(\mathbf{Q}(t)) \leq \quad \mathbb{E}[\textstyle\sum_i \frac{a_i(t)^2+s_i(t)^2}{2} + Q_i(t)(a_i(t) - s_i(t))|\mathbf{Q}(t)].$$

If the RHS of the above inequality is minimized, we have

$$\Delta(\mathbf{Q}(t)) \leq \quad B - \epsilon \textstyle\sum_i Q_i(t),$$

where $\epsilon \geq 0$ is a constant and B is a constant that satisfies

$$B > \mathbb{E}[\sum_i \frac{a_i(t)^2 + s_i(t)^2}{2}|Q_i(t)].$$

Then, based on Theorem 4.1 in [60], the system is stable.

By treating problem (6.12) as a multi-objective (maximizing secure transmission and stabilizing queues) problem and using the penalty method, problem (6.12) is solved by solving the following online problem in each time slot

$$\begin{aligned}
\max \quad & \textstyle\sum_i I_i Q_i s_i + V w_i I_i R_i^s && \text{(6.13a)}\\
s.t. \quad & \forall i, s_i \leq \min(\tau_i \log(1+\gamma_i), Q_i), && \text{(6.13b)}\\
& \forall i, \tau_i \leq 1, && \text{(6.13c)}\\
& \textstyle\sum_i I_i \leq 1, I_i \in \{0,1\}, && \text{(6.13d)}
\end{aligned}$$

where V is a weight assigned to the secure transmission rate, which is used to show the importance of such an objective. For presentation simplicity, the time slot index t is omitted.

Note that the method used here is often referred to as the drift-plus-penalty

method, and the optimality can be guaranteed according to [60]. However, in the system under consideration, due to the subtle difference between the queueing model presented in Sec. 6.3 and [60], the optimality cannot be guaranteed under some circumstances, which will be discussed later. But by using the stochastic network optimization to decompose the problem, it is possible to obtain an online algorithm without the detailed knowledge of the channel information, which is of practical interests.

Algorithm WSSTRM

Define $k_i = s_i/\tau_i$, $g_i(k_i) = (1+\gamma_i)\exp(-k_i) - 1$ and $U_i(k_i) = Q_i k_i + V w_i k_i F(g_i(k_i))$. Problem (6.13) can be reformulated as

$$\begin{aligned} \max \quad & \textstyle\sum_i \tau_i I_i U_i(k_i) && \text{(6.14a)} \\ s.t. \quad & \forall i, k_i \le \log(1+\gamma_i), k_i\tau_i \le Q_i && \text{(6.14b)} \\ & \forall i, \tau_i \le 1, && \text{(6.14c)} \\ & \textstyle\sum_i I_i \le 1, I_i \in \{0,1\}. && \text{(6.14d)} \end{aligned}$$

Note that the above problem is solved by selecting user i^* to transmit, where

$$i^* \in \arg\max_i U_i^*(k_i^*), \tag{6.15}$$

and

$$U_i^*(k_i^*) = \max_{k_i \le \min(Q_i, \log(1+\gamma_i))} U_i(k_i). \tag{6.16}$$

The portion of time user i^* used is $\tau_i^* = 1$.

$U_i^*(k_i^*)$ is obtained by solving (6.16) which might not be a convex problem, since the convexity depends on function F and is generally unknown. But since (6.16) is a one-dimensional problem in a closed set, the optimum solution can be obtained by one-dimensional line search algorithms [7].

In order to perform the line search algorithm efficiently, it is important to know the trend of $U_i(k_i)$, which is critical to the choice of the initial point. Define $\hat{F}(k_i) =$

$F(g_i(k_i))$, and $G(k_i) = k_i\hat{F}(k_i)$. We have

$$\begin{aligned}
G^{'}(k_i) &= k_i\hat{F}^{'}(k_i) + \hat{F}(k_i), \\
G^{''}(k_i) &= k_i\hat{F}^{''}(k_i) + 2\hat{F}^{'}(k_i), \\
g_i^{'}(k_i) &= -1 - g_i(k_i), \\
g_i^{''}(k_i) &= g_i(k_i) + 1, \\
\hat{F}^{'}(k_i) &= F^{'}(g_i(k_i))g_i^{'}(k_i), \\
\hat{F}^{''}(k_i) &= (g_i(k_i)+1)^2(F^{''}(g_i(k_i)) + \frac{F^{'}(g_i(k_i))}{g_i(k_i)+1}).
\end{aligned}$$

Typically, for a wireless channel, the distribution of the SNR has the property: when $\gamma < \gamma_t$, $F^{''}(\gamma) > 0$; when $\gamma > \gamma_t$, $F^{''}(\gamma) < 0$, where γ_t is a SNR threshold. As a result, when $k_i < k_i^{t1}$, $F^{''}(g_i(k_i)) < 0$; when $k_i > k_i^{t1}$, $F^{''}(g_i(k_i)) > 0$. Since $\frac{F^{'}(g_i(k_i))}{g_i(k_i)+1} > 0$ always holds, we have when $k_i < k_i^{t2}$, $\hat{F}^{''}(k_i) < 0$; when $k_i > k_i^{t2}$, $\hat{F}^{''}(k_i) > 0$.

Note that $\hat{F}^{'}(k_i) < 0$, so if $\hat{F}^{''}(k_i) < 0$, then $G^{''}(k_i) < 0$. If $\hat{F}^{''}(k_i) > 0$, then for $k_i < k_i^{t3}$, $G^{''}(k_i) < 0$, and for $k_i > k_i^{t3}$, $G^{''}(k_i) > 0$. In summary, we have that $G^{'}(k_i)$ first decreases and then increases. Since $G^{'}(0) > 0$ and $G^{'}(\log(1+\lambda_i)) < 0$, so $G^{'}(k_i)$ decreases from a positive value to a negative value, and then increases to another negative value. So $G(k_i)$ should be first increasing and then decreasing, the local maximum of $G(k_i)$ is the global maximum, and near the local maximum, $G(k_i)$ is concave.

Since $U_i(k_i) = Q_ik_i + Vw_iG(k_i)$, $U_i(k_i)$ only has three possible trends. First is that $U_i(k_i)$ first increases and then decreases. Second is that it increases. Third is that it has an increase-decrease-increase trend.

Based on the above observation, (6.16) can be solved by finding the first local maximum starting from 0, and comparing it with the boundary value to choose the larger one.

6.5.2 Alternative Relaxed Offline Problem and Optimal Solution

Note that we have show that $R_i^s|_{\tau_i=1}$ first increases and then decreases, and near the maximum of $R_i^s|_{\tau_i=1}$ it is concave. Although this cannot guarantee the objective is concave, as the local maximum of $R_i^s|_{\tau_i=1}$ is also the global maximum, by solving

the following relaxed problem, it yields the maximum secure transmission in the long term.

The relaxed problem is as follows

$$\begin{aligned} \max \quad & \textstyle\sum_i w_i \mathbb{E}[R_i^s(\boldsymbol{\gamma}) I_i(\boldsymbol{\gamma})] && \text{(6.17a)} \\ s.t. \quad & \forall i, s_i(\boldsymbol{\gamma}) \leq \tau(\boldsymbol{\gamma}) \log(1+\gamma_i), && \text{(6.17b)} \\ & \forall i, \tau_i(\boldsymbol{\gamma}) \leq 1. && \text{(6.17c)} \\ & \mathbb{E}[s_i(\boldsymbol{\gamma}) I_i(\boldsymbol{\gamma})] = \lambda_i, && \text{(6.17d)} \\ & \textstyle\sum_i I_i(\boldsymbol{\gamma}) \leq 1, I_i(\boldsymbol{\gamma}) \in \{0,1\}, && \text{(6.17e)} \end{aligned}$$

where $\boldsymbol{\gamma}$ is the instantaneous channel gain vector. The partially augmented Lagrangian dual problem is

$$\begin{aligned} \min_{\mathbf{u}} \max \quad & \textstyle\sum_i (w_i \mathbb{E}[I_i(\boldsymbol{\gamma}) R_i^s(\boldsymbol{\gamma})] + u_i \mathbb{E}[I_i(\boldsymbol{\gamma}) s_i(\boldsymbol{\gamma})] - u_i \lambda_i) && \text{(6.18a)} \\ s.t. \quad & \forall i, s_i(\boldsymbol{\gamma}) \leq \tau(\boldsymbol{\gamma}) \log(1+\gamma_i), && \text{(6.18b)} \\ & \forall i, \tau_i(\boldsymbol{\gamma}) \leq 1, && \text{(6.18c)} \\ & \textstyle\sum_i I_i(\boldsymbol{\gamma}) \leq 1, I_i(\boldsymbol{\gamma}) \in \{0,1\}. && \text{(6.18d)} \end{aligned}$$

Using the primal decomposition, and denoting $k_i(\boldsymbol{\gamma}) = s_i(\boldsymbol{\gamma})/\tau_i(\boldsymbol{\gamma})$, for each $\boldsymbol{\gamma}$, we need to solve the following problem

$$\begin{aligned} \max \quad & \textstyle\sum_i I_i(\boldsymbol{\gamma}) \tau_i(\boldsymbol{\gamma}) k_i(\boldsymbol{\gamma}) (w_i F((1+\gamma_i) e^{-k_i(\boldsymbol{\gamma})} - 1) + u_i) && \text{(6.19a)} \\ s.t. \quad & k_i(\boldsymbol{\gamma}) \leq \log(1+\gamma_i), && \text{(6.19b)} \\ & \forall i, \tau_i(\boldsymbol{\gamma}) \leq 1, && \text{(6.19c)} \\ & \textstyle\sum_i I_i(\boldsymbol{\gamma}) \leq 1, I_i(\boldsymbol{\gamma}) \in \{0,1\}. && \text{(6.19d)} \end{aligned}$$

The optimal solution to the above problem is selecting user $i^*(\boldsymbol{\gamma})$ and use all the time portion for transmission ($\tau^*_{i^*(\boldsymbol{\gamma})}(\boldsymbol{\gamma}) = 1$), where $i^*(\boldsymbol{\gamma}) \in \arg\max_i \tilde{U}_i^*(k_i^*(\boldsymbol{\gamma}))$,

$$\tilde{U}_i^*(k_i^*(\boldsymbol{\gamma})) = \max_{k_i(\boldsymbol{\gamma}) \leq \log(1+\gamma_i)} \tilde{U}_i(k_i(\boldsymbol{\gamma})), \tag{6.20}$$

and

$$\tilde{U}_i(k_i(\boldsymbol{\gamma})) = k_i(\boldsymbol{\gamma})(w_i F((1+\gamma_i) e^{-k_i(\boldsymbol{\gamma})} - 1) + u_i). \tag{6.21}$$

Since $k^*_{i^*(\boldsymbol{\gamma})}(\boldsymbol{\gamma})$ should further satisfy

$$\mathbb{E}[k^*_{i^*(\boldsymbol{\gamma})}(\boldsymbol{\gamma})|i = i^*(\boldsymbol{\gamma})] = \lambda_i, \tag{6.22}$$

and $k^*_i(\boldsymbol{\gamma})$ is a function of u_i, as a result we can obtain u^*_i. A typical algorithm to obtain u^*_i is the gradient descent method, and u_i is updated based on

$$u_i^{(l+1)} = u_i^{(l)} - \epsilon^{(l)}(\mathbb{E}[k_i^{(l)}(\boldsymbol{\gamma})\tau_i^{(l)}(\boldsymbol{\gamma})] - \lambda_i),$$

where $\epsilon^{(l)}$ is a step sequence and square summable [7], and $k_i^{(l)}(\boldsymbol{\gamma})$ and $\tau_i^{(l)}(\boldsymbol{\gamma})$ are the solutions of step l.

Discussion

Noting that u^*_i can be any value as long as $u^*_i + w_i > 0$, this results that the solution to (6.20) is not necessarily always positive. For some $\boldsymbol{\gamma}$ and $\mathbf{u}$, $k^*_{i^*(\boldsymbol{\gamma})}(\boldsymbol{\gamma}) = 0$, which means that the user should not transmit in order to achieve a better secure transmission rate in the long term. However, the online algorithm WSSTRM always selects a user to transmit as long as the user has data to send, and hence it is not always optimal. Comparing function $U_i(k_i)$ in the online algorithm with function $\tilde{U}_i(k_i(\boldsymbol{\gamma}))$ in the offline optimal algorithm, we can see that the purpose of Q_i/V in $U_i(k_i)$ is similar to u_i in $\tilde{U}_i(k_i(\boldsymbol{\gamma}))$ and conceptually Q_i/V can be considered as an "online" Lagrangian multiplier. However, as $Q_i/V \geq 0$ and u^*_i can be negative, conceptually the two algorithms are not identical, if $u^*_i < 0$. As $u^*_i < 0$ only if λ_i is small, which suggests that the algorithm WSSTRM cannot achieve optimality if λ_i is small. The algorithm WSSTRM tries to make a tradeoff between two objectives: maximizing the secure transmission rate and stabilizing the queues in the system. Note that when the arrival rate is small, the requirement for stabilizing the queue becomes less important, as it is possible any resource allocation algorithm can stabilize the queue. Consequently, the scheduler only has one objective: to maximize the secure transmission rate and the algorithm WSSTRM is failed to do so.

6.5.3 Refined Online Algorithm: Algorithm WSSTRM-R

Based on the above analysis, if we can replace Q_i/V by another term which is a more proper "online representation" of u_i, then the resulting algorithm is possible to be an

optimal online algorithm.

Replacing $U_i(k_i)$ in Algorithm WSSTRM by

$$\hat{U}_i(k_i) = (Q_i - Vw_i)k_i + Vw_ik_iF(g_i(k_i)), \tag{6.23}$$

with the same iteration structure as that in Algorithm WSSTRM, we have a queue-length-shifted online algorithm, which is referred to as Algorithm WSSTRM-R later.

Note that Algorithm WSSTRM-R needs to solve

$$\hat{U}_i^*(k_i^*) \quad = \quad \max_{k_i \leq \min(Q_i, \log(1+\gamma_i))} \hat{U}_i(k_i), \tag{6.24}$$

for each user which is slightly different from Algorithm WSSTRM, as the possible increasing trend of $\hat{U}_i(k_i) = (Q_i - Vw_i)k_i + Vw_iG(k_i)$ might be different from $U_i(k_i) = Q_ik_i + Vw_iG(k_i)$, which depends on the value of Q_i. But because $G^{'}(k_i)$ first decreases and then increases, if $Q_i - Vw_i < 0$, then $\hat{U}_i(k_i)$ either decreases or first increases and then decreases. As a result, if $\hat{U}_i^{'}(k_i) < 0$ then the global maximum is achieved at $k_i = 0$, otherwise the algorithm to solve problem (6.24) is identical to the one solving problem (6.16).

Note that Algorithm WSSTRM-R solves the following problem in each time slot:

$$\begin{aligned}
\max \quad & \textstyle\sum_i I_iQ_is_i + Vw_iI_i(R_i^s - s_i) && \text{(6.25a)}\\
s.t. \quad & \forall i, s_i \leq \min(\tau_i \log(1+\gamma_i), Q_i), && \text{(6.25b)}\\
& \forall i, \tau_i \leq 1, && \text{(6.25c)}\\
& \textstyle\sum_i I_i \leq 1, I_i \in \{0,1\}, && \text{(6.25d)}
\end{aligned}$$

which is a decomposed sub-problem of the following problem

$$\begin{aligned}
\max \quad & \lim_{T\to\infty} \tfrac{1}{T} \textstyle\sum_{t=1}^{T} \sum_i w_iI_i(R_i^s(t) - s_i(t)) && \text{(6.26a)}\\
s.t. \quad & \forall i,\ Q_i \text{ is stable}, && \text{(6.26b)}\\
& \forall i,\ s_i(t) \leq \min(\tau_i(t) \log(1+\gamma_i(t)), Q_i(t)), && \text{(6.26c)}\\
& \forall i, \tau_i(t) \leq 1, && \text{(6.26d)}\\
& \textstyle\sum_i I_i(t) \leq 1, I_i(t) \in \{0,1\}. && \text{(6.26e)}
\end{aligned}$$

Due to the stability constraint, we have

$$\lim_{T\to\infty}\frac{1}{T}\sum_{t=1}^{T}\sum_{i}w_iI_i(t)s_i(t)=\sum_{i}w_i\lambda_i, \tag{6.27}$$

which is a constant. As a result, problem (6.26) is equivalent to problem (6.12). Consequently, Algorithm WSSTRM-R can stabilize the system and the performance in terms of maximizing the weighted-sum secure rate should be no worse than Algorithm WSSTRM. Furthermore, comparing $\hat{U}_i(k_i)$ with $\tilde{U}_i(k_i(\boldsymbol{\gamma}))$, the feasible region of $Q_i/V-w_i$ is identical to u_i, and as a result, $Q_i/V-w_i$ can be considered as a proper "online representation" of u_i. In Sec. 6.7 we will show that indeed Algorithm WSSTRM-R is an online optimal algorithm.

6.6 Case Study: Eavesdropper with an AWGN channel

6.6.1 Algorithm WSSTRM

If the eavesdropper has an AWGN channel without fading, the secure transmission rate of user i in time slot t becomes

$$\begin{aligned}R_i^s(t) &= s_i(t)\delta(r_i(t)-s_i(t)-C^e(t)),\\ &= s_i(t)\delta([\log(\frac{1+\gamma_i(t)}{1+\gamma_e(t)})]^+-\frac{s_i(t)}{\tau_i(t)}),\end{aligned}$$

where $\delta(x)$ is an indicator function. $\delta(x)=1$ if $x\geq 0$ and $\delta(x)=0$ if otherwise.

Thus we have

$$U_i(k_i)=Q_ik_i+Vw_ik_i\delta([\log(\frac{1+\gamma_i}{1+\gamma_e})]^+-k_i). \tag{6.28}$$

Denote $R_e^i=\max(\log(1+\gamma_i)-\log(1+\gamma_e),0)$, which is the maximal supported secure data size[2] of user i that does not lead to secrecy outage. The transmission

[2] R_e^i can also be viewed as the maximal supported secure rate, as we assume the slot length is one second.

strategy for user i is as follows

$$U_i^*(k_i^*) = (Q_i + Vw_i)k_i^* \tag{6.29}$$

if $Q_i \leq R_e^i$ or $\min(\frac{Q_i}{Vw_i}\log(1+\gamma_e), \frac{Q_i^2}{Q_i+Vw_i}) \leq R_e^i \leq Q_i$, where $k_i^* = \min(Q_i, R_e^i)$; otherwise

$$U_i^*(k_i^*) = Q_i k_i^*, \tag{6.30}$$

where $k_i^* = \min(Q_i, \log(1+\gamma_i))$.

The above transmission strategy can be explained as follows. When the available data (Q_i) is smaller than the maximal supported secure data size (R_e^i), the user should use all the resources to transmit all the available data, and the data are fully protected by the physical-layer encoder. If the SNR of the user is larger than a threshold, the maximal supported secure data size is chosen and all the data are fully protected by the physical-layer encoder; if the SNR of the user is worse, then the user should use all the available resource or transmit all the available data, but the data are not fully protected and secrecy outage happens with probability one.

6.6.2 Algorithm WSSTRM-R

Similar to Algorithm WSSTRM, the transmission strategy for user i is as follows:

$$U_i^*(k_i^*) = Q_i k_i^* \tag{6.31}$$

if $Q_i \leq R_e^i$ or $(Q_i - Vw_i)\min(\frac{\log(1+\gamma_e)}{Vw_i}, 1) \leq R_e^i \leq Q_i$, where $k_i^* = \min(Q_i, R_e^i)$; otherwise

$$U_i^*(k_i^*) = (Q_i - Vw_i)k_i^*, \tag{6.32}$$

where $k_i^* = \min(Q_i, \log(1+\gamma_i))$.

Comparing Algorithm WSSTRM-R with Algorithm WSSTRM we can find that the key difference lies in a threshold and such difference results in that the long-term secure transmission rate can be improved when λ_i is small, *i.e.,* Q_i is small due to Little's law. If λ_i is small, $Q_i - Vw_i < 0$ should almost always hold. Consequently the data transmitted are always fully protected and the secure transmission rate is identical to λ_i. While for Algorithm WSSTRM, when λ_i is small but $Q_i > R_e^i$,

whether the data can be fully protected also depends on the channel condition of user i, and the transmitted data are not always fully protected and thus cannot be optimal.

6.6.3 Offline Problem and Analysis

Similarly, we can obtain the solution to the relaxed offline problem as in Sec. 6.5.2, and the transmission strategy for each user depends on the Lagrangian multiplier u_i^* and is shown as follows.

Case 1 when $u_i^* > 0$: if $\gamma_i \geq (1+\gamma_e)^{1+u_i^*/w_i} - 1$, then $U_i^*(k_i^*) = (u_i^* + w_i)k_i^*(\gamma_i)$ and $k_i^*(\gamma_i) = R_e^i$; otherwise $U_i^*(k_i^*) = u_i^* k_i^*(\gamma_i)$ and $k_i^*(\gamma_i) = \log(1+\gamma_i)$.

Case 2 when $u_i^* = 0$: if $\gamma_i \geq \gamma_e$ then $U_i^*(k_i^*) = w_i k_i^*(\gamma_i)$ and $k_i^*(\gamma_i) = R_e^i$; otherwise $U_i^*(k_i^*) = 0$ and $k_i^*(\gamma_i) \in [0, \log(1+\gamma_i)]$.

Case 3 when $-w_i < u_i^* < 0$: if $\gamma_i \geq \gamma_e$ then $U_i^*(k_i^*) = (u_i^* + w_i)k_i^*(\gamma_i)$ and $k_i^*(\gamma_i) = R_e^i$; if $\gamma_i \leq \gamma_e$ then $U_i^*(k_i^*) = 0$ and $k_i^*(\gamma_i) = 0$.

Case 4 when $u_i^* \leq -w_i$: $U_i^*(k_i^*) = 0$ and $k_i^*(\gamma_i) = 0$.

Furthermore, the Lagrangian multiplier is determined by the arrival rate $\boldsymbol{\lambda}$ through (6.22).

First we can see that Case 4 should never happen as using this transmission strategy cannot achieve the rate stability. Second, if $\gamma_i \geq (1+\gamma_e)^{1+[u_i^*]^+} - 1$, then the strategy is to always transmit data in R_e^i to achieve the maximal secure transmission rate. Third, if $\gamma_i < (1+\gamma_e)^{1+[u_i^*]^+} - 1$, then depending on the value of u_i^*, the strategy decides whether to transmit and how many data to transmit. If $u_i^* \geq 0$ then the user transmits at a positive rate in order to achieve the rate stability; if $u_i^* < 0$, then the user does not transmit as the rate-stable condition can be satisfied by the transmission strategy when $\gamma_i \geq \gamma_e$, which implies that the traffic load should be small. Fourth, by comparing the offline optimal transmission strategy with the online algorithm, the key difference is how to transmit data if the legitimate receiver's channel is bad. The online algorithm always tries to empty the queue, while the offline strategy will stop transmission if the traffic load is small, which utilizes the information about the traffic load explicitly.

In order to analyze the property of the solution, we restrict our attention to the single legitimate receiver case.

Denote

$$\begin{aligned}
\lambda_i^{\text{th1}} &= \int_{\gamma_e}^{\infty} \log(\frac{1+\gamma_i}{1+\gamma_e}) f_i(\gamma_i) d\gamma_i, \\
\lambda_i^{\text{th2}} &= \mathbb{E}[\log(1+\gamma_i)] - \log(1+\gamma_e)[1-F(\gamma_e)], \\
\bar{R}_i &= \mathbb{E}[\log(1+\gamma_i)], \\
\bar{R}_i^{\text{th}}(\gamma) &= \int_0^{\gamma} \log(1+\gamma_i) f_i(\gamma_i) d\gamma_i, \\
R_e &= \log(1+\gamma_e).
\end{aligned}$$

after some calculations, the maximal secure transmission rate can be obtained as:

if $\lambda_i \leq \lambda_i^{\text{th1}}$,

$$R_i^{s*} = \lambda_i, \tag{6.33}$$

and $u_i^* < 0$; if $\lambda_i^{\text{th1}} \leq \lambda_i \leq \lambda_i^{\text{th2}}$,

$$R_i^{s*} = \lambda_i^{\text{th1}}, \tag{6.34}$$

$u_i^* = 0$; if $\lambda_i \geq \lambda_i^{\text{th2}}$,

$$R_i^{s*} = \lambda_i - \bar{R}_i^{\text{th}}(\gamma_{\text{th}}) \tag{6.35}$$

where γ_{th} is the solution to $\frac{\bar{R}_i - \lambda_i}{R_e} = 1 - F(\gamma_{\text{th}})$ and $u_i^* > 0$.

As the secrecy outage probability for a single user is $\bar{P}_i^{\text{out}} = \frac{\lambda_i - R_i^{s*}}{\lambda_i}$, we have

$$\bar{P}_i^{\text{out}} = \begin{cases} 0, & \lambda_i \leq \lambda_i^{\text{th1}}, \\ 1 - \frac{\lambda_i^{\text{th1}}}{\lambda_i}, & \lambda_i^{\text{th1}} \leq \lambda_i \leq \lambda_i^{\text{th2}} \\ \frac{\bar{R}_i^{\text{th}}(\gamma_{\text{th}})}{\lambda_i}, & \lambda_i^{\text{th2}} \leq \lambda_i. \end{cases}$$

From the above equation we can see that when the arrival rate is small, the user does not experience secrecy outage, as the arrival traffic lies inside the secrecy capacity region. When the arrival rate further increases, the secrecy outage probability also increases, but with two different increasing speeds, related to the arrival rate.

6.7 Evaluation and Discussion

6.7.1 Simulation Setting

In the simulation, we consider a system that contains N legitimate receivers and one eavesdropper. Although the number of eavesdropper is limited to one, it is sufficient to quantify the performance of the proposed algorithms and investigate the relationship between the system performance and different network configurations. The channel gains of the receivers and the eavesdropper are modeled as Nakagami fadings. So, γ_i and γ_e are Gamma distributed random variables. The probability density function of γ_i is

$$f(x) = (\frac{m_i}{\bar{\gamma}_i})^{m_i} \frac{x^{m_i-1}}{\Gamma(m_i)} \exp(-\frac{m_i x}{\bar{\gamma}_i}), \quad (m_i \geq 0.5),$$

and the CDF of γ_i is

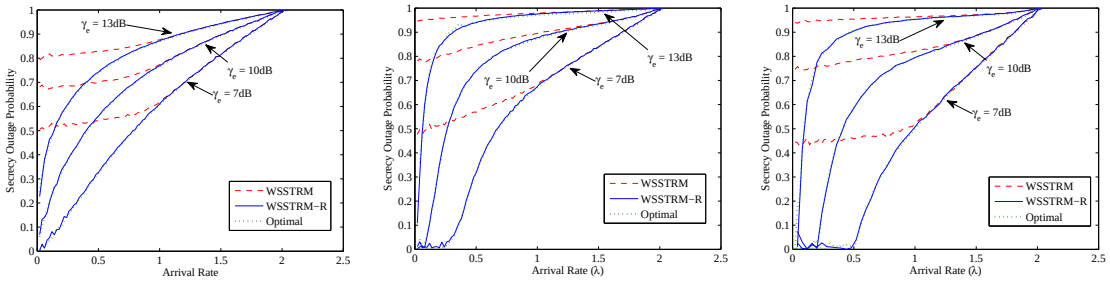
$$F(x) = \frac{\int_0^{m_i x/\bar{\gamma}_i} t^{m_i-1} e^{-t} dt}{\Gamma(m_i)}, \quad (m_i \geq 0.5),$$

where m_i is the fading parameter of user i, and $\bar{\gamma}_i$ is the average channel gain of user i. Note that, m_i is used to control the variability of γ_i, and a small m_i results in a large variation of γ_i. When $m_i = 1$, the Nakagami fading becomes a Rayleigh fading. When $m_i \to \infty, \gamma_i = \bar{\gamma}_i$, the channel becomes an AWGN channel.

The amount of traffic arrival in each time slot $a_i(t)$ is a Poission random variable, and the system frequency bandwidth is normalized to 1. So the units of the secure transmission rate and the arrival rate are both bps/Hz and are omitted hereinafter. We choose the parameter V as 100. During the simulation, we have run a sufficient large number of time slots in order to ensure that the system converges to its steady state, and the results are collected from the steady state. For each simulation setting, we repeat ten times and take the average. Other parameters used for different network configurations are listed in the caption of each figure.

6.7.2 Single Legitimate Receiver

We assume that the legitimate receiver experiences Rayleigh fading ($m_i = 1$) with mean SNR as 10dB ($\bar{\gamma}_i = 10$dB). By changing the channel setting for the eavesdropper and the arrival rate of the data for the legitimate receiver we can investigate the



(a) $m_e = 1$ (b) $m_e = 10$ (c) $m_e = \infty$

Figure 6.3: Secrecy outage probability, single legitimate receiver, $\bar{\gamma}_i = 10$dB, $m_i = 1$

performance of the secrecy outage probability.

When the eavesdropper experiences a Rayleigh fading channel ($m_e = 1$), the corresponding secrecy outage probability is illustrated in Fig. 6.3-a. With the increase of the arrival rate λ, the secrecy outage probability increases. However, with different $\bar{\gamma}_e$, *i.e.*, the SNR of the eavesdropper's channel, the increasing speed is different. When $\bar{\gamma}_e = 7$dB, the secrecy outage probability is roughly linear with λ. With a large $\bar{\gamma}_e$, when λ is small, the secrecy outage probability increases quickly w.r.t. λ, and a small increment of λ results in a large secrecy outage probability increase.

When the eavesdropper experiences Nakagami fading with $m_e = 10$, the results are shown in Fig. 6.3-b. A similar trend as in Fig. 6.3-a can be observed. But note that when $\bar{\gamma}_e$ is small and λ is also small, the secrecy outage probability is almost zero and is not related to λ.

Fig. 6.3-c illustrates the secrecy outage probability when the eavesdropper experiences an AWGN channel without fading ($m_e = \infty$). Similar to Fig. 6.3-b, we can see when λ is small, the system is able to achieve zero secrecy outage, which confirms the analysis in Sec. 6.6.

From all the above three figures we can see that, Algorithm WSSTRM cannot achieve the optimal secrecy outage probability when the arrival rate is small, which validates our analysis in Sec. 6.5, as when the arrival rate is small, Q_i/Vw_i is not a proper "online representative" of the Lagrangian multiplier. But when the arrival rate is large, Algorithm WSSTRM can achieve the optimal secrecy outage probability, as under this circumstance Q_i/Vw_i can properly represent the Lagrangian multiplier as it is positive. Further note that the curves of Algorithm WSSTRM-R are always overlapped with the curves of the optimal results, which indicates that Algorithm WSSTRM-R is optimal.

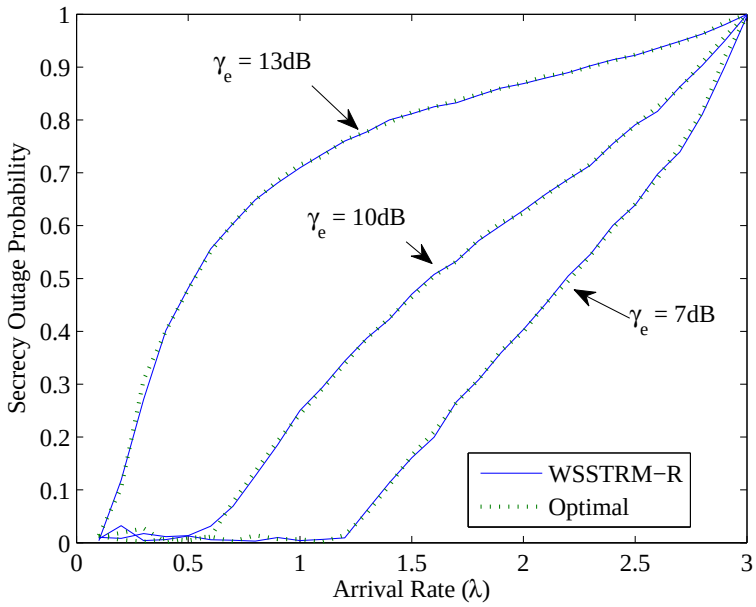


Figure 6.4: Secrecy outage probability, multiple legitimate receivers, $\bar{\gamma}_i = 10$dB, $m_i = 1$, $m_e = \infty$

6.7.3 Multiple Legitimate Receivers

As Algorithm WSSTRM cannot achieve optimality even in a system with a single legitimate receiver, in this subsection, we only discuss Algorithm WSSTRM-R and the optimal results, showing that in the multiple legitimate receivers case, Algorithm WSSTRM-R is also optimal.

During the simulation, we use $N = 5$, $\boldsymbol{\lambda} = [1, 2, 3, 4, 5]/15 \times \lambda$, where λ is the aggregated arrival rate. We assume that all legitimate receivers experience Rayleigh fading ($m_i = 1$) and have identical $\bar{\gamma}_i = 10$dB. The result that when the eavesdropper experiences an AWGN channel is illustrated in Fig. 6.4. Firstly, comparing with Fig. 6.3-c, the trend in Fig. 6.3-c is preserved in the multiple legitimate receivers case. Furthermore, in the multiple legitimate receivers case, the secrecy outage probability is smaller than that in the single legitimate receiver case when the system is subject to the same arrival rate, because of the capacity increasing thanks to the multi-user diversity. Secondly, the curve of Algorithm WSSTRM-R is overlapped with that of the optimal result suggests that Algorithm WSSTRM-R is an optimal online algorithm in the multi-legitimate-receiver case, and is able to stabilize the queues in the system, achieve reliable communication and minimize the secrecy outage probability.

6.8 Conclusions

In this chapter, we investigated the secrecy outage probability in multiuser wireless systems with stochastic traffic. We defined the secrecy outage probability in a system with channel-adaptive transmissions and discussed how to minimize it subject to the communication reliability and queue stability constraints. Stochastic network optimization framework has been used to decompose the problem into an online problem, and an online algorithm WSSTRM was proposed. We further discussed an alternative offline problem and based on the study of the offline problem, we found that the first proposed online algorithm may not be optimal. Motivated by this, we proposed a refined online algorithm WSSTRM-R. Furthermore, We discussed and analyzed the transmission strategy if the eavesdropper experiences an AWGN channel and further compared the proposed algorithms. Simulation results confirmed that when the traffic load is small, Algorithm WSSTRM is not optimal, but the algorithm WSSTRM-R is indeed optimal. Furthermore, several observations were obtained on the relationship between the secrecy outage probability of the system and traffic load, channel conditions, etc. These observations provide important insights and guidelines for the design and resource management of future wireless systems using secure communication technologies.

Chapter 7

Conclusions and Further Research Issues

7.1 Conclusions

In this dissertation, we have discussed various aspects of scheduling and resource allocation in multi-user wireless systems.

1. We have discussed the resource allocation problem in a saturated multi-user wireless system using superposition coding to fully explore the capacity of the broadcast channel, and proposed scheduling algorithms in a practical system with hierarchical modulation and demonstrated the remarkable performance improvement.

2. The stability regions of utility-based opportunistic scheduling algorithms in a multi-user wireless system with stochastic traffic are derived and the structure properties have been discussed. The results show that the stability region might be non-convex which is harmful for the operation of wireless systems.

3. The limiting properties of an overloaded multi-user wireless system with throughput-optimal scheduling algorithm have been quantified and the corresponding throughput is analyzed. The results can be used to quantify the transient system performance in a temporary overloaded system which is of practical interests.

4. Secrecy outage probability in a multi-user wireless system has been investigated through a resource allocation problem and we have proposed two optimal algorithms, one online and one offline. We have further investigated the impact

of channel condition and arrival rate on the secrecy outage probability which can be further used as design guideline in a system using information-theoretic security technologies.

7.2 Further Research Issues

There are many open issues beckon for further research in the topics we discussed in this dissertation.

1. For the work on the scheduling in a SPC/HM-aided wireless system, we list the possible further research issues as follows. First, the algorithm proposed in Chapter 2 can be easily extended to a system with parallel GBC, by adding an iterative multi-user water-filling algorithm associating the power allocation to each parallel channel, as in [82, 61]. This will further increase the computational complexity and the low complexity solution should be investigated in the future. This work also opens up many cross-layer problems beckoning for further research, e.g., how to optimize resource allocation with SPC considering the application layer traffic characteristics and multicast scenarios [41, 104, 40]. Second, although CSI feedback in PFS has been well investigated in [17], the discussion should be extended to the multi-user case, as in SPC/HM-aided wireless system, multiple users may get scheduled in each time slot. Third, the performance of the heuristic algorithm for J-layer HM case discussed in Chapter 3 should be further investigated. As the optimal algorithm for J-layer HM requires to search for all the possible combination, which leads to a high computational complexity, it is important to study the tradeoff between the throughput gain and the computational complexity. Fourth, how to extend the solution in Chapter 3 to a coded HM-based system and to consider HARQ requires further investigation. A possible approach is by discretizing the power allocation to constrain the number of transmission modes (*i.e.*, modulation and coding schemes given the power allocation) then obtaining a SNR-transmission mode mapping. However, this approach cannot be applied to the case with coded modulation (such as trellis-coded modulation) directly.

2. For the work on the stability region on the opportunistic scheduling, we have the following directions should be studied in the future. First, what is the impact of ϵ when updating the smoothed rate measurement. If ϵ is not proper, the

smoothed rate measurement may not be able to converge, especially if the flow exhibits a bursty feature. Thus the rate allocation is not stationary, and the resultant impact on the stability region is unclear. Second, given the average arrival rate, how to design the weight to each user to stabilize the system and how the designed weight affects the system performance needs to be investigated further.

3. Regarding the limiting properties in overloaded wireless systems with throughput-optimal scheduling, there are several open issues left behind. First, we have assumed the achievable rate region $\mathcal{C}(t)$ is always an N-dimension region. But in practice, if the channel is deep faded, the corresponding user may have zero achievable rate, and we cannot always increase the allocated rate of one user by decreasing the rates of other users. Second, the throughput-optimal scheduling considered is only queue-length based, which excludes the delay-driven throughput-optimal scheduling. Given the linear relationship between delay and queue length[3], the analysis for delay-driven throughput-optimal may have a similar result. Third, some queue-length based throughput-optimal scheduling algorithms, such as EXP-rule [73], are excluded from the discussion. How to extend our work to consider a more general throughput-optimal scheduling, such as the one proposed in [105] remains an open issue.

4. In Chapter 6, we have shown that directly apply stochastic network optimization framework may not lead to the optimal results, since the queue length might not be a proper representation of the Lagrangian multiplier. However, we have also shown that by reformulating the original problem, stochastic network optimization framework can be directly used and leads to the optimal results. It is important to revisit the stochastic network optimization to understand this interesting behavior, which might improve the theory and lead to more applications.

Bibliography

[1] A. Agustin, J. Vidal, and O. Muoz. Performance of downlink schedulers with superposed or orthogonal transmissions. In *IEEE International Conference on Communications (ICC)*, May 2010.

[2] M. Andrews. Instability of the proportional fair scheduling algorithm for hdr. *IEEE Transactions on Wireless Communcations*, 3(5):1422–1426, Sep 2004.

[3] M. Andrews, K. Kumaran, K. Ramanan, A.L. Stolyar, R. Vijayakumar, and P. Whiting. Scheduling in a queueing system with asynchronously varying service rates. *Probability in the Engineering and Informational Sciences*, 18:191–217, 2004.

[4] D. P. Bertsekas. *Nonlinear Programming : 2nd Edition.* Athena Scientific, 1999.

[5] M. Bloch, J. Barros, M. R. D. Rodrigues, and S. W. McLaughlin. Wireless information-theoretic security. *IEEE Transactions on Information Theorys*, 54(6):2515–2534, June 2008.

[6] T. Bonald and J. Roberts. Performance modeling of elastic traffic in overload. *ACM SIGMETRICS Performance Evaluation Review*, 29(1):342–343, June 2001.

[7] S. P. Boyd and L. Vandenberghe. *Convex optimization.* Cambridge University Press, 2004.

[8] L. Cai, S. Xiang, Y. Luo, and J. Pan. Scalable modulation for video transmission in wireless networks. *IEEE Transactions on Vehicular Technology*, 60(9):4314–4323, Nov 2011.

[9] C. W. Chan, M. Armony, and N. Bambos. Fairness in overloaded parallel queues. *arXiv preprint arXiv:1011.1237v2 (2011).*

[10] T. Cover. Broadcast channels. *IEEE Transactions on Information Theory*, 18(1):2–14, Jan 1972.

[11] R. Egorova, S. Borst, and B. Zwart. Bandwidth-sharing networks in overloaded. *Performance Evaluation*, 64(9):978–993, Oct. 2007.

[12] A. Eryilmaz, R. Srikant, and J.R. Perkins. Stable scheduling policies for fading wireless channels. *IEEE/ACM Transactions on Networking*, 13(2):411–424, April 2005.

[13] P. C. Fishburn. *Utility theory for decision making*. Robert E. Krieger Publishing Company, Inc, 1970.

[14] S. Floyd and V. Jacobson. Random early detection gateways for congestion avoidance. *IEEE/ACM Transactions on Networking*, 1(4):397–413, Aug. 1993.

[15] L. Georgadis. Lexicographically optimal balanced networks. In *IEEE INFOCOM*, 2001.

[16] L. Georgiadis and L. Tassiulas. Optimal overload response in sensor networks. *IEEE Transactions on Information Theory*, 52(6):2684–2696, June 2006.

[17] D. Gesbert and M.-S. Alouini. How much feedback is multi-user diversity really worth? In *IEEE International Conference on Communications (ICC) 2004*, Jun 2004.

[18] S. Goel and R. Negi. Secret communication in presence of colluding eavesdroppers. In *IEEE Military Communications Conference*, 2005.

[19] A. Goldsmith. *Wireless Communications*. Cambridge University Press, 2005.

[20] S. Gollakota and D. Katabi. Physical layer wireless security made fast and channel independent. In *IEEE INFOCOM*, 2011.

[21] P. K. Gopala, L. Lai, and H. El-Gamal. On the secrecy capacity of fading channels. *IEEE Transactions on Information Theory*, 54(10):4687–4698, Oct. 2008.

[22] O. Gungor, J. Tan, C. E. Koksal, H. El-Gamal, and N. B. Shroff. Joint power and secret key queue management for delay limited secure communication. In *IEEE INFOCOM*, 2010.

[23] H. Guo, H. Hu, Y. Zhang, and H.-H. Chen. On stability regions in opportunistic scheduled-packet access networks. *IEEE Transactions on Vehicular Technology*, 59(1):295306, Jan 2010.

[24] Y. Guo, E. Lefeber, Y. Nazarathy, G. Weiss, and H. Zhang. Stability and performance for multi-class queueing networks with infinite virtual queues. *Queueing Systems*, 2013.

[25] M. Hossain, M.-S. Alouini, and V. Bhargava. Rate adaptive hierarchical modulation-assisted two-user opportunistic scheduling. *IEEE Transactions on Wireless Communcations*, 6(6):2076–2085, Jun 2007.

[26] M. Hossain, M.-S. Alouini, and V. Bhargava. Two-user opportunistic scheduling using hierarchical modulations in wireless networks with heterogenous average link gains. *IEEE Transactions on Communcations*, 58(3):880–889, Mar 2010.

[27] J. Huang, V. G. Subramanian, R. Agrawal, and R. A. Berry. Downlink scheduling and resource allocation for ofdm systems. *IEEE Transactions on Wireless Communcations*, 8(1):288–296, Jan 2009.

[28] J. Huang, V. G. Subramanian, R. Agrawal, and R. A. Berry. Joint scheduling and resource allocation in uplink ofdm systems for broadband wireless access networks. *IEEE Journal on Selected Areas in Communications*, 27(2):226–234, Feb 2009.

[29] Y. Hwang and H. C. Papadopoulos. Physical-layer secrecy in awgn via a class of chaotic ds/ss systems: analysis and design. *IEEE Transactions on Signal Processing*, 52(9):2637–2649, Sept. 2004.

[30] K. Jagannathan, M. Markakis, E. Modiano, and J. N. Tsitsiklis. Queue-length asymptotics for generalized max-weight scheduling in the presence of heavy-tailed traffic. *IEEE/ACM Transactions on Networking*, 20(2):1096–1111, Apr. 2012.

[31] R. Jain, D.-M. Chiu, and W. R. Hawe. A quantitative measure of fairness and discrimination for resource allocation in shared computer system. *DEC Research Report TR-301*, 1984.

[32] A. Jalali, R. Padovani, and R. Pankaj. Data throughput of cdma-hdr a high efficiency-high data rate personal communication wireless system. In *IEEE 51st Vehicular Technology Conference Proceedings*, 2000.

[33] A. Jdidi and T. Chahed. Impact of hierarchical modulation on proportional fair in ofdma-based networks. In *International Conference on Communications and Networking (ComNet) 2010*, Nov 2010.

[34] N. Jindal, S. Vishwanath, and A. Goldsmith. On the duality of gaussian multiple-access and broadcast channels. *IEEE Transactions on Information Theory*, 50(5):768–783, May 2004.

[35] F. Kelly, A. Maulloo, and D. Tan. Rate control for communication networks: shadow prices, proportional fairness and stability. *Journal of the Operational Research Society*, 49(3):237–252, Mar 1998.

[36] A. Khisti, A. Tchamkerten, and G. W. Wornell. Secure broadcasting over fading channels. *IEEE Transactions on Information Theory*, 54(6):2453–2469, June 2008.

[37] H. Kim and Y. Han. A proportional fair scheduling for multicarrier transmission systems. *IEEE Communications Letters*, 9(3):210–212, Mar 2005.

[38] J. Kim, J. Lee, K. Son, S. Song, and S. Chong. Two-hop opportunistic scheduling in cooperative cellular networks. *IEEE Transactions on Vehicular Technology*, 61(11):4194–4199, Nov 2012.

[39] R. Knopp and P. A. Humblet. Information capacity and power control in single-cell multiuser communications. In *IEEE International Conference on Communications*, volume 1, pages 331–335, Jun 1995.

[40] M. Kobayashi, H. Nakayama, N. Ansari, and N. Kato. Reliable application layer multicast over combined wired and wireless networks. *IEEE Transactions on Multimedia*, 11(8):1466–1477, Dec 2009.

[41] M. Kobayashi, H. Nakayama, N. Ansari, and N. Kato. Robust and efficient stream delivery for application layer multicasting in heterogeneous networks. *IEEE Transactions on Multimedia*, 11(1):166–176, Jan 2009.

[42] V. G. Kulkarni. Fluid models for single buffer systems. *Frontiers in queueing*, pages 321–338, 1998.

[43] H. J. Kushner and P. A. Whiting. Convergence of proportional-fair sharing algorithms under general conditions. *IEEE Transactions on Wireless Communcations*, 3(4):1250–1259, Jul 2004.

[44] H. J. Kushner and G. G. Yin. *Stochastic Approximation and Recursive Algorithms and Applications, 2nd.* Springer-Verlag, 2003.

[45] S. K. L.-Y.-Cheong and M. E. Hellman. The gaussian wire-tap channel. *IEEE Transactions on Information Theory*, 24(4):451–456, July 1978.

[46] E. Leonardi, M. Mellia, F. Neri, and M. A. Marsan. On the stability of input-queued switches with speed-up. *IEEE/ACM Transactions on Networking*, 9(1):104–118, Feb. 2001.

[47] E. Leonardi, M. Mellia, F. Neri, and M.A. Marsan. Bounds on delays and queue lengths in input-queued cell switches. *Journal of the ACM*, 50(4):520–550, Jul 2003.

[48] T. Li, J. Ren, Q. Ling, and A. Jain. Physical layer built-in security analysis and enhancement of cdma systems. In *IEEE Military Communications Conference*, 2005.

[49] Y. Liang, H. V. Poor, and S. Shamai (Shitz). Information theoretic security. *Foundations and Trends in Communications and Information Theory*, 5(4-5):355–580, Jun. 2009.

[50] Y. Liang, H. V. Poor, and L. Ying. Wireless broadcast networks: Reliability, security, and stability. In *Information Theory and Applications Workshop*, 2008.

[51] H. Liu, J. Yang, Y. Wang, and Y. Chen. Collaborative secret key extraction leveraging received signal strength in mobile wireless networks. In *IEEE INFOCOM*, 2012.

[52] Y. Liu, K. Lau, O. Takeshita, and M. Fitz. Optimal rate allocation for superposition coding in quasi-static fading channels. In *IEEE International Symposium on Information Theory 2002*, 2002.

[53] Y. Liu and P. Ning. Enhanced wireless channel authentication using time-synched link signature. In *IEEE INFOCOM*, 2012.

[54] Z. Mao, C. E. Koksal, and N. B. Shroff. Towards achieving full secrecy rate in wireless networks: A control theoretic approach. In *Information Theory and Applications Workshop*, 2011.

[55] M. G. Markakis, E. Modiano, and J. N. Tsitsiklis. Max-weight scheduling in queueing networks with heavy-tailed traffic. *IEEE/ACM Transactions on Networking*, 22(1):257–270, Feb. 2014.

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

[57] S. Meyn. Stability and asymptotic optimality of generalized maxweight policies. *SIAM Journal on Control and Optimization*, 47:32593294, 2009.

[58] J. Mo and J. Walrand. Fair end-to-end window-based congestion control. *IEEE/ACM Transactions on Networking*, 8(5):556–567, Oct 2000.

[59] M. J. Neely. Order optimal delay for opportunistic scheduling in multi-user wireless uplinks and downlinks. *IEEE/ACM Transactions on Networking*, 16(5):1188–1199, Oct. 2008.

[60] M. J. Neely. Stochastic network optimization with application to communication and queueing systems. *Synthesis Lectures on Communication Networks*, 2010.

[61] C. Ng, K. Shum, C. Sung, and T. Lok. A layered decomposition framework for resource allocation in multiuser communications. *IEEE Transactions on Vehicular Technology*, 60(2):729–733, Feb 2011.

[62] F. Oggier and B. Hassibi. The secrecy capacity of the mimo wiretap channel. *IEEE Transactions on Information Theory*, 57(8):4961–4972, Aug. 2011.

[63] N. Patwari and S. K. Kasera. Robust location distinction using temporal link signatures. In *ACM MobiCom'07*, 2007.

[64] D. Qiao, M. C. Gursoy, and S. Velipasalar. Secure wireless communication and optimal power control under statistical queueing constraints. *IEEE Transactions on Information Forensics and Security*, 6(3):628–639, Sept. 2011.

[65] B. Radunovic and J.-Y. Le Boudec. A unified framework for max-min and min-max fairness with applications. *IEEE/ACM Transactions on Networking*, 15(5):1073–1083, Oct. 2007.

[66] R. T. Rockafellar. *Convex Analysis*. Princeton University Press, 1970.

[67] B. Rong and A. Ephremides. Joint mac and rate control for stability and delay in wireless multi-access channels. *Performance Evaluation*, 68(8):658669, Aug 2011.

[68] B. Sadiq, S. J. Baek, and G. de Veciana. Delay-optimal opportunistic scheduling and approximations: the log rule. *IEEE/ACM Transactions on Networking*, 19(2):405–418, April 2011.

[69] B. Sadiq and G. de Veciana. Large deviations sum-queue optimality of a radial sum-rate monotone opportunistic scheduler. *IEEE Transactions on Information Theory*, 56(7):3395–3412, July 2010.

[70] S. Sadr, A. Anpalagan, and K. Raahemifar. Radio resource allocation algorithms for the downlink of multiuser ofdm communication systems. *IEEE Communications Surveys and Tutorials*, 11(3):92–106, Sep. 2009.

[71] K. Seong, R. Narasimhan, and J. M. Cioffi. Queue proportional scheduling via geometric programming in fading broadcast channels. *IEEE Journal on Selected Areas in Communications*, 24(8):1593–1602, Aug. 2006.

[72] D. Shah and D. Wischik. Fluid models of congestion collapse in overloaded switched networks. *Queueing Systems*, 69(2):121–143, Oct. 2011.

[73] S. Shakkottai and A. L. Stolyar. Scheduling for multiple flows sharing a time-varying channel: the exponential rule. *American Mathematical Society Translations, Series 2*, 2000.

[74] M. Sharif, A.F. Dana, and B. Hassibi. Differentiated rate scheduling for gaussian broadcast channels. In *IEEE International Symposium on Information Theory*, Sep 2005.

[75] A. Sheikholeslami, D. Goeckel, H. P.-Nik, and D. Towsley. Physical layer security from inter-session interference in large wireless networks. In *IEEE INFOCOM*, 2012.

[76] Y-S Shiu, S-Y Chang, H-C Wu, S. C-H Huang, and H-H Chen. Physical layer security in wireless networks: a tutorial. *IEEE Wireless Communications*, 18(2):66–74, April 2011.

[77] A. L. Stolyar. On the asymptotic optimality of the gradient scheduling algorithm for multiuser throughput allocation. *Operations Research*, 53(1):12–25, Jan 2005.

[78] A. L. Stolyar. Large deviations of queues sharing a randomly time-varying server. *Queueing Systems*, 59(1):135, Jan 2008.

[79] Digital Video Broadcasting (DVB) Framing Structure. Channel coding and modulation for digital terrestrial television (dvb-t) v1.5.1. ETSI Standard ETS 300 744, Nov. 2004.

[80] L. Tassiulas and A. Ephremides. Dynamic server allocation to parallel queues with randomly varying connectivity. *IEEE Transactions on Information Theory*, 39(2):466–478, Mar 1993.

[81] S. Tekin, S. Andradttir, and D. G. Down. Dynamic server allocation for unstable queueing networks with flexible servers. *Queueing Systems*, 70:45–79, 2012.

[82] D. N. Tse. Optimal power allocation over parallel gaussian broadcast channels. Technical report.

[83] D. N. Tse. Optimal power allocation over parallel gaussian broadcast channels. In *IEEE International Symposium on Information Theory 1997*, Jun 1997.

[84] V. J. Venkataramanan and X. Lin. On wireless scheduling algorithms for minimizing the queue-overflow probability. *IEEE/ACM Transactions on Networking*, 18(3):788801, June 2010.

[85] P. Viswanath, D. Tse, and R. Laroia. Opportunistic beamforming using dumb antennas. *IEEE Transactions on Information Theory*, 48(6):1277–1294, Jun 2002.

[86] P. K. Vitthaladevuni and M.-S. Alouini. A closed-form expression for the exact ber of generalized pam and qam constellations. *IEEE Transactions on Communications*, 52(5):698–700, May 2004.

[87] Q. Wang, H. Su, K. Ren, and K. Kim. Fast and scalable secret key generation exploiting channel phase randomness in wireless networks. In *IEEE INFOCOM*, 2011.

[88] X. Wang and L. Cai. Limiting properties of overloaded multiuser wireless systems with throughput-optimal scheduling. *IEEE Transactions on Communications, 2014*.

[89] X. Wang and L. Cai. Proportional fair scheduling in hierarchical modulation aided wireless networks. *IEEE Transactions on Wireless Communications*, 12(4):1584–1593, Apr 2013.

[90] X. Wang and L. Cai. Resource allocation in a k-user wireless broadcast system with n-layer superposition coding. In *Wireless Communications and Networking Conference (WCNC), 2013 IEEE*, pages 789–794, April 2013.

[91] X. Wang and L. Cai. Stability region of opportunistic scheduling in wireless networks. *IEEE Transactions on Vehicular Technology*, XX(XX):XX, Jan 2014.

[92] X. Wang, W. Chen, and Z. Cao. Sparc: Superposition-aided rateless coding in wireless relay systems. *IEEE Transactions on Vehicular Technology*, 60(9):4427–4438, Nov 2011.

[93] X. Wang, Y. Chen, L. Cai, and J. Pan. Secrecy outage probability in multiuser wireless systems with stochastic traffic. *Submitted to IEEE Transactions on Wireless Communications, 2014*.

[94] X. Wang, Y. Chen, L. Cai, and J. Pan. Scheduling in a secure wireless network. In *IEEE INFOCOM*, 2014.

[95] X. Wang and G. B. Giannakis. Resource allocation for wireless multiuser ofdm networks. *IEEE Transactions on Information Theory*, 57(7):4359–4372, Jul 2011.

[96] X. Wang, G. B. Giannakis, and A. G. Marques. A unified approach to qos-guaranteed scheduling for channel-adaptive wireless networks. *Proceedings of the IEEE*, 95(12):2410–2431, Dec 2007.

[97] G. Wunder and T. Michel. Optimal resource allocation for parallel gaussian broadcast channels: Minimum rate constraints and sum power minimization. *IEEE Transactions on Information Theory*, 53(12):4817–4822, Dec 2007.

[98] A. Wyner. The wire-tap channel. *Bell System Technical Journal*, 54(8):1355–1387, Oct. 1975.

[99] L. Xu, X. Shen, and J. W. Mark. Dynamic fair scheduling with qos constraints in multimedia w-cdma cellular networks. *IEEE Transactions on Wireless Communications*, 3(1):60–73, Jan 2004.

[100] L. Yang and L. Hanzo. A recursive algorithm for the error probability evaluation of m-qam. *IEEE Communications Letters*, 4(10):304–306, Oct 2000.

[101] Z. Yang, L. Cai, Y. Luo, and J. Pan. Topology-aware modulation and error-correction coding for cooperative networks. *IEEE Journal on Selected Areas in Communications*, 30(2):379–387, Feb 2012.

[102] E. M. Yeh and A. S. Cohen. Delay optimal rate allocation in multiaccess fading communications. In *42nd Allerton Conf. on Communication, Control, and Computing*, 2004.

[103] J. Zhang, M. H. Firooz, N. Patwari, and S. K. Kasera. Advancing wireless link signatures for location distinction. In *ACM MobiCom'08*, 2008.

[104] R. Zhang, R. Ruby, J. Pan, L. Cai, and X. Shen. A hybrid reservation/contention-based mac for video streaming over wireless networks. *IEEE Journal on Selected Areas in Communications*, 28(3):389–398, Apr 2010.

[105] C. Zhou and G. Wunder. A fundamental characterization of stability in broadcast queueing systems. In *IEEE International Symposium on Information Theory*, 2009.

[106] C. Zhou and G. Wunder. Throughput-optimal scheduling with low average delay for cellular broadcast systems. *EURASIP Journal on Advances in Signal Processing*, 2009.

[107] X. Zhou, M. R. McKay, B. Maham, and A. Hjrungnes. Rethinking the secrecy outage formulation: A secure transmission design perspective. *IEEE Communications Letters*, 15(3):302–304, March 2011.