Stability and Performance of Emerging Wireless Networks

by

Zhefeng Jiang

A dissertation submitted to the Graduate Faculty of
Auburn University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Auburn, Alabama
August 6, 2016

Keywords: Cloud offloading, Full-duplex, Femtocell, Cognitive radio, LTE-unlicensed, Lyapunov optimization

Approved by

Shiwen Mao, Ginn Professor and WEREC Director of Electrical and Computer Engineering
Prathima Agrawal, Samuel Ginn Distinguished Professor of Electrical and Computer Engineering
Jitendra K Tugnait, James B Davis Professor of Electrical and Computer Engineering
Erkan Nane, Associate Professor of Mathematics and Statistics

Abstract

In this work, we analysis the application of emerging wireless communications on the stability of computing and transmission queues of mobile devices. Firstly, we present a Lyapunov optimization-based scheme for cloud offloading scheduling, as well as download scheduling for cloud execution output, for multiple applications running in a mobile device with a multi-core CPU. We derive an online algorithm and prove performance bounds for the proposed algorithm with respect to average power consumption and average queue length. which is indicative of delay, and reveal the fundamental trade-off between the two optimization goals.

Extending Long Term Evolution (LTE) to unlicensed bands, termed LTE-unlicensed promises tremendous spectrum to meet the increasing wireless data transmission demands and we proposed a novel distributed online algorithm for opportunistic sharing of unlicensed bands among LTE-unlicensed base stations (BS), while guaranteeing the QoS of user equipments (UE). We first derive a Lyapunov optimization based algorithm for BS's to evaluate the true value of unlicensed spectrum, guarantee a maximum delay, and minimize the packet drop rate. We then develop a distributed auction mechanism to maximize the social welfare in each auction and enable optimal spectrum reuse. We prove that BS's bid truthfully with the proposed algorithm, while UEs' QoS requirements on delay and packet drop rate can be guaranteed with bounded optimality gaps. We also reveal an interesting trade-off between delay and packet drop rate.

Full-duplex is gaining significant interest recently and can double the system throughput theoretically. In this work, we investigate the trade-off between energy consumption and delay in a multi-channel full-duplex wireless LAN (WLAN). The goal is to minimize the energy consumption while keeping the packet queues stable. With Lyapunov optimization, we develop an online scheme to achieve the goals with optimized channel assignment, transmission scheduling, and transmission mode selection. We prove the optimality of the proposed algorithm and derive upper

bounds for the average queue length and energy consumption, which demonstrate the energy-delay trade-off.

We finally studied the problem of joint access control and spectrum resource allocation in a two-tier femtocell network with one macro base station (MBS) and multiple Femto Access Points (FAP). The objective is to maximize the overall network capacity, while guaranteeing the quality of service (QoS) requirement of all UE. We develop an access scheme for Macro User Equipments (MUE) and a spectrum allocation mechanism for the FAPs. Spectrum allocation is employed as an incentive mechanism to encourage FAPs to serve more MUEs. We also derive an upper bound of the network-wide capacity through a reformulation of the problem.

Acknowledgments

Over the past four years I have received support and inspiration from a great number of individuals. Firstly, I would like to express the deepest appreciation to my committee chair Dr. Shiwen Mao. He has been a mentor, colleague, and friend. His support and guidance has made this an exiting and rewarding journey. I would like to thank my committee of Dr. Prathima Agrawal, Dr. Jitendra K Tugnait and Dr. Erkan Nane for their knowledge and support in each step I move from ideas to a complete work. In addition, I would like to thank Dr. David A. Umphress for his efforts and time in improving this work.

Special thanks: this work was supported in part by the US National Science Foundation (NSF) under Grants CNS-0953513 and CNS-1247955, and through the Wireless Engineering Research and Education Center (WEREC) at Auburn University.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Recent years have witnessed the exceptional increase of mobile devices, including smartphones and tablets. In the US, the number of smartphone users has been steadily increasing for some years and forecasts estimate that the increase of smartphone users in North American will continue rising steadily into the future. The prediction of the number of smartphone users in the United States from 2010 to 2019 is shown as Fig.1.1. For 2016, the number of smartphone users is estimated to reach 207.2 million in the United States and is estimated to exceed 2 billion worldwide by that time [1]. Accompany with the fast increasing of the mobile device users, the number of apps available on mobile devices is also expanding steadily. For Apple along, there are 1.5 million Apps available in June, 2015, and the number of available apps in the Apple App Store from July 2008 to June 2015 is shown in Fig.1.2. With the burst of applications targeting mobiles devices, mobile devices are expected to be capable of running multiple applications simultaneously and take part of the role of a laptop, such as mobile office, online videos and video games, which requires strong computational capacity and high speed wireless data transmission.

However, due to the mobility requirement, the energy supply and physical size of mobile devices are limited, the computational capacity of mobile devices can hardly been met. Under such circumstances, smart phone manufacturers are keep adopting stronger CPUs which always come with thermal problems, heavier batteries and less the battery time. In other words, it is challenging to balance the demands for stronger computation capacity and the mobility of mobile devices in the foreseeing future. Mobile cloud offloading has been recognized as an effective solution to the limited resource problem [4, 5]. Mobile cloud offloading involves wireless communication, cloud computing and mobile computing, which brings rich computation and storage resource of cloud computing providers to resource-constraint mobile devices through wireless channel of Internet.

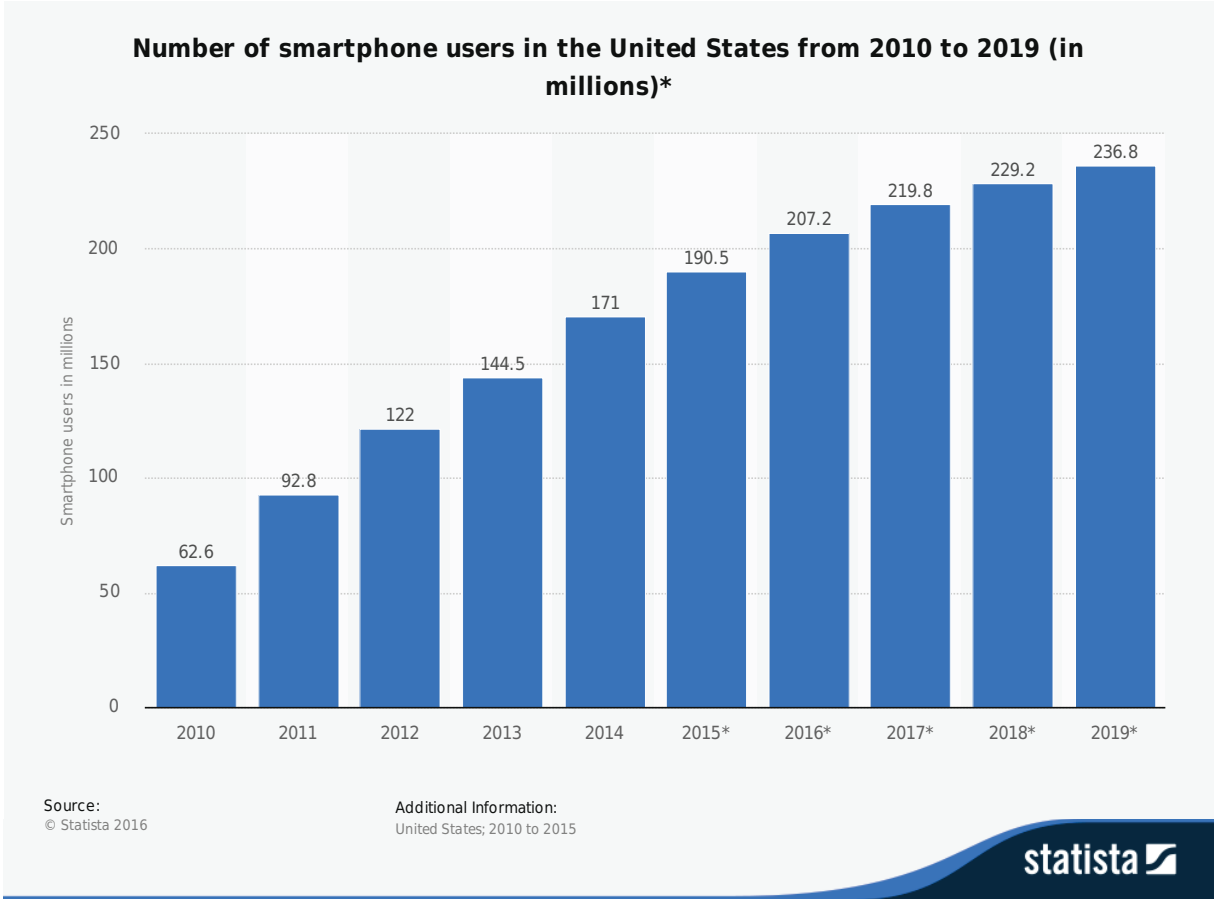

Figure 1.1: Prediction of the number of smartphone users in the United States from 2010 to 2019 (in millions) [1]

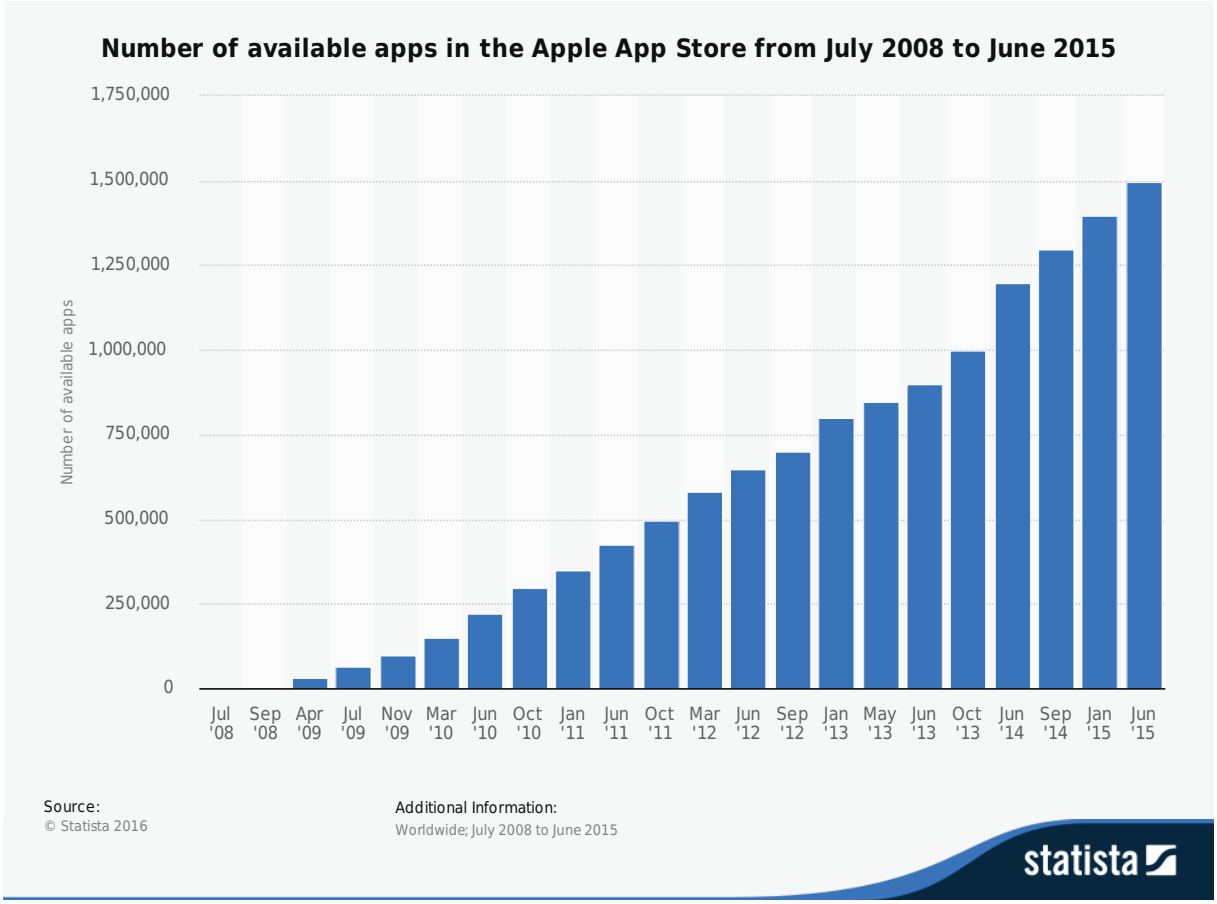

Figure 1.2: number of available apps in the Apple App Store from July 2008 to June 2015 [2]

With offloading, we can store our photos and videos in the cloud and fetch it whenever it is needed. Furthermore, computation intensive tasks can also be offloaded to software clones in the cloud [7], so that most computation can be executed in the cloud to greatly reduce the burden on the mobile device [8]. However, offloading data and computational tasks could involve considerable communications between mobile devices and cloud clones, which could consume a large amount of energy and incur extra delay. Hence, the decision between cloud offloading or local execution should be carefully made at each mobile device, taking into account the energy consumption and delay of various options, as well as the status of the wireless network.

To support the higher speed of wireless data transmission of mobile devices, full-duplex radio, LTE-unlicensed and Femtocells were introduced to increase the wireless link capacity.

To meet the so-called 1000x mobile data challenge [87], extending LTE to the unlicensed spectrum, as specified in LTE Rel-10 – Rel-13 [83, 84], has recently gained significant attention [83, 84, 87, 88, 90, 92–98]. However, there are two main challenges to the success of the so-called *LTE-unlicensed* technology. First, the unlicensed bands are already occupied by many existing wireless networks (e.g., WiFi). It is essential to enable the coexistence of LTE-unlicensed with existing unlicensed band users, i.e., to avoid significant performance degradation to existing users while achieving high capacity gains with LTE-unlicensed. Second, the interference in unlicensed bands is unpredictable, which is detrimental to the performance of LTE-unlicensed users. Hence, it is important to effectively manage the interference between LTE-unlicensed and existing users, and that among LTE-unlicensed users themselves. In this work, we investigate the problem of opportunistic spectrum sharing among LTU-unlicensed BS's. We consider the License Assisted Access (LAA) scenario, in which licensed and unlicensed carrier bands are integrated and used [84]. We also adopt the LBT mechanism for co-existence of LTE-unlicensed and WiFi [95]. For the LTE-unlicensed BS's deployed in the same area on both licensed and unlicensed bands, we propose a novel distributed online algorithm for opportunistic sharing of unlicensed bands among the BS's, while guaranteeing the QoS of UEs in the form of bounded worst case delay and minimized packet drop rate.

Through effective self-interference cancellation, full-duplex transmission, i.e., transmitting and receiving simultaneously in the same band, has been successfully demonstrated [36]. With various self-interference cancellation techniques, full-duplex transmission has the potential to increase and even double the wireless link capacity [37]. Due to imperfect self-interference cancellation, the residual self-interference may still lead to a lower signal-to-interference-plus-noise ratio (SINR) and deteriorate the performance of a full-duplex link [42]. Additional power is needed to combat the residual self-interference to achieve a suitable SINR. As a result, full-duplex transmission may not always be helpful, and there is a trade-off between the energy cost and delay in the design of full-duplex wireless networks [43].

Femtocells, also named as Femto Access Points (FAP), are small, low power cellular base stations (BS). Femtocells are designed for use at homes and small enterprises, and are usually connected to the core network with broadband wireline connections [50]. In addition to providing a shortcut to the core network, the wireline connection also enables coordinations among FAPs and macrocell base stations (MBS) to improve the performance of the two-tier network. Femtocells are considered as a low-cost and effective solution to extend wireless coverage and offload voice and wireless data. This is really important, as research indicates that 70% of data traffic take place indoor where the coverage of conventional cellular networks is usually poor. With femtocells, the distance between BS and a User Equipments (UE) is greatly reduced, thus enabling better signal transmissions and better spatial reuse of spectrum. In this work, we investigate the problem of access control and spectrum resource allocation in two-tier femtocell networks. We assume one MBS and multiple FAPs in the area and consider the open access scheme. The FUEs are always connected to the corresponding FAPs, while the MUEs can choose between the MBS and a nearby FAP for connection. The spectrum is divided into two parts, one for the MBS and the other part for the FAPs. To provide incentives to FAPs for serving MUEs, we allow dynamic partition of the spectrum according to the network dynamics; more bandwidth will be allocated to the FAPs if they serve more MUEs.

The contributions of this work are summarized as follows.

- We present a Lyapunov optimization-based scheme for cloud offloading scheduling, as well as download scheduling for cloud execution output, for multiple applications running in a mobile device with a multi-core CPU. We derive an online algorithm and prove performance bounds for the proposed algorithm with respect to average power consumption and average queue length, which is indicative of delay, and reveal the fundamental trade-off between the two optimization goals. The performance of the proposed online scheduling scheme is validated with trace-driven simulations.

- We proposed a novel distributed online algorithm for opportunistic sharing of unlicensed bands among LTE-unlicensed base stations (BS), while guaranteeing the QoS of UE. We first derive a Lyapunov optimization based algorithm for BS's to evaluate the true value of unlicensed spectrum, guarantee a maximum delay, and minimize the packet drop rate. We then develop a distributed auction mechanism to maximize the social welfare in each auction and enable optimal spectrum reuse. We prove that BS's bid truthfully with the proposed algorithm, while UEs' QoS requirements on delay and packet drop rate can be guaranteed with bounded optimality gaps. We also reveal an interesting trade-off between delay and packet drop rate. The proposed algorithm is validated with simulations.

- We investigate the trade-off between energy consumption and delay in a multi-channel full-duplex WLAN. The goal is to minimize the energy consumption while keeping the packet queues stable. With Lyapunov optimization, we develop an online scheme to achieve the goals with optimized channel assignment, transmission scheduling, and transmission mode selection. We prove the optimality of the proposed algorithm and derive upper bounds for the average queue length and energy consumption, which demonstrate the energy-delay trade-off. The proposed algorithm is validated with simulations.

- We study the problem of joint access control and spectrum resource allocation in a two-tier femtocell network with one MBS and multiple FAP. The objective is to maximize the overall network capacity, while guaranteeing the QoS requirement of all UE. We develop an access scheme for MUE and a spectrum allocation mechanism for the FAPs. Spectrum allocation is employed as an incentive mechanism to encourage FAPs to serve more MUEs. We also derive an upper bound of the network-wide capacity through a reformulation of the problem.

Chapter 2

Energy Delay Trade-off in Cloud Offloading for Mutli-core Mobile Devices

2.1 Introduction

There is a proliferation of mobile devices in recent years, such as smartphones and tablets, which are becoming more and more powerful with even multi-core CPUs. However, mobile devices still suffer from comparably limited resources. For example, the power of a smartphone comes at the cost of higher burden on the battery. As a result, although we are freed from a wireline data connection, we are still highly dependent on a power socket and charger. In addition, smartphones usually have relatively limited storage. With many apps, photos, and multimedia files recorded or cached, the internal storage space of our mobile devices can be easily depleted.

Cloud offloading has been recognized as an effective solution to the limited resource problem [4, 5]. With offloading, we can store our photos and videos in the cloud and fetch it whenever it is needed. Furthermore, computation intensive tasks can also be offloaded to software clones in the cloud [7], so that most computation can be executed in the cloud to greatly reduce the burden on the mobile device [8]. However, offloading data and computational tasks could involve considerable communications between mobile devices and cloud clones, which could consume a large amount of energy and incur extra delay. Hence, the decision between cloud offloading or local execution should be carefully made at each mobile device, taking into account the energy consumption and delay of various options, as well as the status of the wireless network.

In this chapter, we study the problem of effective cloud offloading scheduling while considering downloading the output of cloud execution, for mobile devices with muti-core CPUs. We also consider task scheduling among the multiple cores of the CPU and frequency adaptation for the CPU, considering both energy cost and user experience with respect to delay. Specifically,

there are several trade-offs in making the optimal decisions. First, cloud offloading involves data transmissions from the mobile device to the cloud, as well as downloading the output of cloud execution, through a stochastic and thus unpredictable wireless channel. The energy efficiency of cloud offloading could be poor when the wireless coverage is weak. In such cases, energy may be conserved if we delay cloud offloading and downloading until the channel gets better, but at the cost of additional delays. Furthermore, cloud offloading may not be a good choice for applications with a large amount of offloading data to be sent to the cloud, or a large amount of output data to be downloaded after cloud execution, since transmitting the data over a wireless channel may consume considerable power and incur large delay as well, which offset the gains achieved by executing the task in the cloud. Similarly, energy can be conserved for local execution by reducing the CPU frequency, but at the cost of slower execution (and thus increased delay) of the tasks.

Motivated by these observations, we present a holistic formulation of the problem of optimal cloud offloading decision making for multiple applications running in a multi-core mobile device. The formulation takes into account the above trade-offs by incorporating the key control knobs, including CPU frequency and computation capability at the mobile device, offloading and downloading data volume of the applications, and the time-varying capacity and expected offloading power consumption of the wireless connection.

We then develop an effective solution algorithm to the formulated problem. The proposed scheduling algorithm is based on the Lyapunov optimizing framework [9, 14, 46]. It dynamically schedules the tasks in the task queues for cloud offloading or local execution, downloads output from the cloud for offloaded tasks, and in the case of local execution, tunes the CPU frequency to balance energy consumption and delay, based on the current network condition and task queue backlogs. The proposed algorithm is inherently an *online algorithm*, meaning that it does not require information about the stationary distributions of the arrival and wireless channel processes, neither does any future application and network state information. It makes decisions based on the current queue backlogs and wireless channel conditions. Such an online algorithm would be useful

for real-time applications. We derive upper bounds on the average energy consumption and average queue length achieved by the proposed algorithm, which clearly reveal the trade-off between energy consumption and delay in optimal cloud offloading. The proposed algorithm is validated with trace-driven simulations, where the mobile device has both LTE and WiFi connections, and the energy-delay trade-off is clearly revealed.

The rest of this chapter is organized as follows. The system model and problem statement are presented in Section 2.2. The proposed algorithm is developed in Section 2.3 and evaluated with trace-driven simulations in Section 2.4. We review related work in Section 2.5. Section 2.6 concludes the chapter. The main notation used in this chapter is summarized in Table 2.1 and Table 2.2.

Table 2.1: Notation

Symbol	Description
N	number of applications
$\mathcal{N}$	set of applications
N'	number of applications can be offloaded
$\mathcal{N}'$	set of applications can be offloaded
$\mathcal{Q}(t)$	set of application queues
$Q_i(t)$	queue of application i
$A_i(t)$	new arrivals to queue i at time t
$\mathcal{A}(t)$	set of arrivals at time t
λ_i	arrival rate of application i
$\vec{\lambda}$	set of arrival rate
$B_i(t)$	number of tasks of application i executed locally at time t
$B_i^O(t)$	number of executed tasks of application i downloaded at time slot t
$B_i^D(t)$	service rate of the cloud output queue for application i at time slot t
$\theta_i(k)$	computational complexity of task k of application i
$D_i(k)$	data size for offloading task k of application i
$D_i^D(k)$	data size of cloud execution output of task k of application i
$Q_i^D(t)$	returned output queue at of application i at the end of time slot t
$\mathcal{Q}^D(t)$	set of returned output queue at of application at the end of time slot t
$A_i^D(t)$	arrival to queue $Q_i^D(t)$ at time slot t
$\mathcal{A}_i(t)$	set of arrival to queues $Q^D(t)$ at time slot t
$f(t)$	clock frequency of CPU at time slot t
v	voltage of the mobile CPU at time t
η'	energy coefficient of CPU
$\varepsilon_i(t)$	energy consumption of core i at time slot t
$\varepsilon(t)$	overall energy consumption of the CPU at time t
$\alpha^L(t)$	set of application being executed locally
$\Theta_i(t)$	amount of computations a CPU core can offer to application i
M	number of CPU core
η	adjusted energy coefficient
$\omega_O(t)$	uplink wireless data rate
$\omega_D(t)$	downlink wireless data rate
$\alpha^O(t)$	offloaded application at time t
$p_O(t)$	energy consumption of offloading
$\alpha^D(t)$	application downloaded for execution data at time t

Table 2.2: Notation(contd.)

Symbol	Description
$p_D(t)$	energy consumption of downloading at time t
$\bar{P}$	average overall power consumption
$P(t)$	overall power consumption at time slot t
$\bar{Q}$	average overall queue length, including task queues and downloading queues
$L(Q(t))$	Lyapunov function
V_p	Lyapunov constant
P^{opt}	optimum (minimum) energy consumption
$\epsilon > 0$	distance between the data arrival rate vector $\vec{\lambda}$ and the system capacity region under the proposed algorithm
ξ	defined in (2.39), (2.41) and (2.42)
φ	a term defined in (2.30)

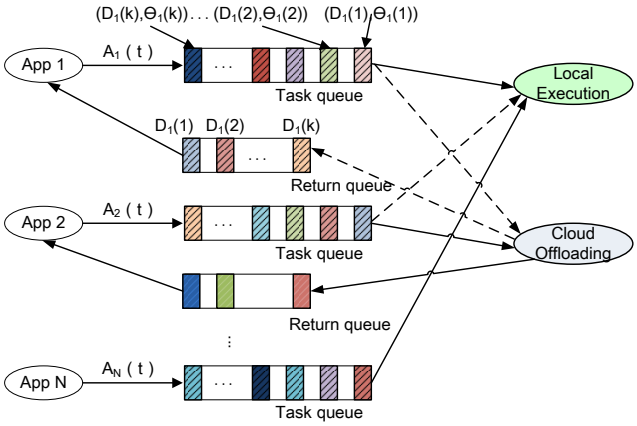

Figure 2.1: The system model.

2.2 System Model and Problem Statement

2.2.1 System Model

The system model is illustrated in Fig. 2.1. We consider a mobile device having N applications running,[1] denoted as $\mathcal{N} = \{1, 2, \cdots, N\}$, among which $1 \leq N' \leq N$ applications, denoted as $\mathcal{N}'$, can be offloaded to the cloud. The tasks generated from each application are enqueued and processed in a First-In-First-Out (FIFO) manner. In addition, we assume that the arrival and execution of these tasks follow a discrete, time-slotted system. In particular, the queue of tasks waiting to be processed for application i at the beginning of time slot t is denoted as $Q_i(t)$, and the overall queue lengths at the beginning of time slot t are denoted as

$$\mathcal{Q}(t) = \{Q_1(t), Q_2(t), \cdots, Q_N(t)\}. \tag{2.1}$$

In time slot t, the tasks generated by applications are denoted as

$$\mathcal{A}(t) = \{A_1(t), A_2(t), \cdots, A_N(t)\}, \tag{2.2}$$

[1] A multiple-thread application that enables parallel computing, can be treated as multiple applications.

which can be regarded as new arrivals to $\mathcal{Q}(t)$. In this chapter, we assume that each $A_i(t)$ is independent and identically distributed (i.i.d.) over time slots and the expectations of them, i.e., the average arrival rates, are denoted as

$$\vec{\lambda} \triangleq \mathbb{E}\{\mathcal{A}(t)\} = \{\lambda_1, \lambda_2, \cdots, \lambda_N\}. \tag{2.3}$$

The departing tasks from queue $\mathcal{Q}(t)$ at time slot t is either scheduled for local execution, denoted as

$$\mathcal{B}(t) = \{B_1(t), B_2(t), \cdots, B_N(t)\}, \tag{2.4}$$

or offloaded to the cloud, denoted as

$$\mathcal{B}^O(t) = \{B_1^O(t), B_2^O(t), \cdots, B_N^O(t)\}. \tag{2.5}$$

In addition, we assume that for task k of application i, the computational complexity for local execution, $\theta_i(k)$ (i.e., the amount of computations required to accomplish the task), the data size for offloading, $D_i(k)$ (i.e., the amount of data transmitted for executing the task in the cloud), and the data size of the cloud execution output, $D_i^D(k)$ (i.e., the results to be returned to the mobile device), are all i.i.d. random variables. If the task cannot be offloaded to the cloud, then we have $D_i(k) = \infty$ and $D_i^D(k) = 0$. Alternatively, if the task can only be offloaded to the cloud, then we have $\theta_i(k) = \infty$.

When a task is offloaded, it is first processed by a server in the cloud and then the output of cloud execution is returned to the mobile device. Hence, there is also a queue for the output data of cloud execution (e.g., at the access point or base station). Let $\mathcal{Q}^D(t)$ denote the returned output queue at the end of the time slot t, as shown in Fig. 2.1. We have

$$\mathcal{Q}^D(t) = \{Q_1^D(t), Q_2^D(t), \cdots, Q_N^D(t)\}, \tag{2.6}$$

where $Q_i^D(t) = 0$ for $i \in \mathcal{N} \backslash \mathcal{N}'$, as there will be no output from cloud computing if the task cannot be offloaded. The arrival to the queue $\mathcal{Q}^D(t)$ can be denoted as

$$\mathcal{A}^D(t) = \{A_1^D(t), A_2^D(t), \cdots, A_N^D(t)\}, \tag{2.7}$$

for an application i task that is to be offloaded, $|A_i^D(t)| = |B_i^O(t)|$. That is, if we ignore the time a cloud server takes to process the task, there is an increment of queue length in $Q_i^D(t)$ if a task in $Q_i(t)$ is offloaded to the cloud.

2.2.2 Local Execution Energy Consumption Model

For applications that are executed locally at the mobile device, most of the energy consumption comes from the CPU and the screen. As the screen energy consumption is largely dependent on the user habit, we do not take this part into account in this chapter.[2] The energy consumption is thus mainly determined by the CPU operation.

In particular, the CPU energy consumption is proportional to v^2, where v is the CPU voltage [10]. Furthermore, the clock frequency of the CPU at time slot t, denoted as $f(t)$, is shown approximately linear to the CPU voltage v [10]. Therefore, the CPU power consumption in a CPU core occupied by application i in time slot t can be approximated as

$$\varepsilon_i(t) = \eta' \cdot f_i^2(t), \tag{2.8}$$

where η' is the energy coefficient determined by the CPU hardware architecture. As the energy consumption is linear with $f^2(t)$, energy can be saved by reducing the CPU frequency, which, however, will slow down the execution of the tasks.

A CPU schedule can be represented by $\{\alpha^L(t), \Theta(t)\}$, where $\alpha^L(t) \in \mathcal{N}$ is the set of applications being executed locally, $\Theta(t) = \{\Theta_1(t), \Theta_2(t), \cdots, \Theta_N(t)\}$, and $\Theta_i(t)$ is the amount of

[2] It may be annoying to dynamically adjust the display size, resolution, or brightness during the execution of an application. We simply assume some constant amount of energy consumption associated with this part.

computations a CPU core can offer to application i at time slot t. Note that $\Theta_i(t) = 0$, if $i \notin \alpha^L(t)$. Assuming that there are M cores in the CPU. We have $|\alpha^L(t)| \leq M$, i.e., the number of parallel computing applications cannot exceed the number of cores in the CPU. For a given CPU architecture, the computational capability $\Theta_i(t)$ is usually linear with the CPU frequency. Hence, the CPU energy consumption at time t is also a quadratic function of $\Theta_i(t)$, i.e.,

$$\varepsilon_i(t) = \eta \cdot \Theta_i^2(t), \tag{2.9}$$

where η is the adjusted energy coefficient. The total energy consumption for local execution is

$$\varepsilon(t) = \sum_{i=1}^{N} \varepsilon_i(t). \tag{2.10}$$

2.2.3 Offloading Energy Consumption Model

For applications that can be offloaded to the cloud, we make the following assumptions. First, we assume that a software clone has already been associated with each application in the cloud to support cloud computing [11], such that only the latest use generated data, application status updates, and cloud execution output, refereed to as *offloading data*, need to be transmitted between the mobile device and the cloud.

Second, we focus on the channel models associated with the wireless interfaces and ignore the delay and energy consumption in the cloud, which are justifiably minor issues comparing to that on the mobile device side. It is typical for a smartphone to choose one of the mobile networks (e.g., 2G, 3G, LTE, and WiFi) and the corresponding data rate is determined by the operator and the baseband chip configuration. We adopt the network selection algorithm proposed in [12] to choose between a cellular network and WiFi, and focus on the task scheduling problem in this chapter.

Let $\omega_O(t)$ be the wireless link data rate from the mobile device to the cloud, and $\omega_D(t)$ the data rate from the cloud to the mobile device. An offloading decision is denoted as

$$\alpha^O(t) \in \{\mathcal{N}', ' \text{idle}'\}. \tag{2.11}$$

That is, the device can choose to offload a task from one of the eligible queues or remain idle (i.e., to choose local execution). Then, the expected energy consumption is denoted as $p_O(t)$. Similarly, the decision for downloading the cloud execution output can be denoted as

$$\alpha^D(t) \in \{\mathcal{N}', ' \text{idle}'\}, \tag{2.12}$$

and the expected energy consumption is denoted as $p_D(t)$.

2.2.4 Queuing And The Overall Energy Consumption Model

As discussed, energy can be conserved by optimizing the execution decision for the application tasks, i.e., local execution or offloading to the cloud. For local execution, energy can be saved by reducing the CPU frequency (i.e., running the application at a lower speed, which leads to a smaller $\Theta(t)$). For offloading, energy can be saved by only using good channels for transmission of offloading data and receiving the cloud output. There maybe an additional delay to wait for the channel to get better. If we aggressively save power by these means, the applications will suffer from large delays; the lengths of the task queues may increase to very high levels and the system may become unstable. We need to balance energy saving and delay, which is indicated by the task queue length.

Define the total power consumption in time slot t as

$$P(t) = \varepsilon(t) + p_O(t) + p_D(t). \tag{2.13}$$

Based on the local execution and offloading energy consumption models, the overall energy consumption of the mobile device can be derived as follows.

$$\begin{aligned}\bar{P} &\triangleq \limsup_{T\to\infty} \frac{1}{T}\sum_{t=0}^{T-1} P(t) \\ &= \limsup_{T\to\infty} \frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}\{\varepsilon(t) + p_O(t) + p_D(t)\}.\end{aligned} \tag{2.14}$$

We define the average task and output queue length, denoted as $\bar{Q}$, for evaluation of the energy-queue trade-off as

$$\bar{Q} \triangleq \limsup_{T\to\infty} \frac{1}{T}\sum_{t=0}^{T-1}\sum_{i=1}^{N} \mathbb{E}\{Q_i(t) + Q_i^D(t)\}, \tag{2.15}$$

where $Q_i(t)$ is the task queue length for application i at time t, and $Q_i^D(t)$ is the cloud output queue length for application i at time t. We consider the system to be stable if the average queue length is bounded, i.e., the limit in (2.15) exists.

The dynamics of the task queue backlog $Q_i(t)$ can be written as

$$Q_i(t+1) = \max\{Q_i(t) + A_i(t) - B_i(t) - B_i^O(t), 0\}, \forall\, i, \tag{2.16}$$

where $B_i(t)$ is the service rate at time t defined as follows.[3]

$$B_i(t) = \begin{cases} \arg\max_{\{b\}} \left\{\sum_{k=1}^{b} \theta_i(k) \le \Theta_i(t)\right\}, \\ \qquad\qquad \text{if } i \in \alpha^L(t) \\ \arg\max_{\{b\}} \left\{\sum_{k=1}^{b} D_i(k) \le \omega_O(t)\right\}, \\ \qquad\qquad \text{if } i \in \alpha^O(t) \\ 0, \qquad\qquad \text{otherwise.} \end{cases} \tag{2.17}$$

[3]We assume that the duration of a time slot is large enough such that any task can be executed locally, offloaded to the cloud, or with output downloaded from the cloud in less than one time slot. This can be achieved by choosing a suitable time slot duration or by partitioning big tasks into smaller ones.

Note that $\alpha^L(t)$ and $\alpha^O(t)$ should not point to the same application i, as it is inefficient to both offload and locally execute the same application task at the same time. If $i \in \alpha^L(t)$, the task queue of application i is executed locally and $B_i(t)$ is the maximum number of tasks can be executed locally at time slot t. If $i \in \alpha^O(t)$, the tasks of application i are offloaded to the cloud and $B_i(t)$ is the maximum number of tasks can be offloaded at this time slot.

Similarly, the dynamics of the cloud output queue backlog $Q_i^D(t)$ can be written as

$$Q_i^D(t+1) = \max\{Q_i^D(t) + A_i^D(t) - B_i^D(t), 0\}, \forall\, i \in \mathcal{N}', \tag{2.18}$$

where $\left|A_i^D(t)\right| = \left|B_i^O(t)\right|$ and $B_i^D(t)$ is the service rate at time i for the cloud output queue defined as

$$B_i^D(t) = \begin{cases} \arg\max_{\{b\}} \left\{\sum_{k=1}^{b} D_i^D(k) \leq \omega_D(t)\right\}, \\ \qquad\qquad \text{if } i \in \alpha^D(t) \\ 0, \qquad\qquad \text{otherwise.} \end{cases} \tag{2.19}$$

If $i \in \alpha^D(t)$, the cloud output queue i is downloaded and $B^D(t)$ is the maximum number of tasks that can download their cloud output at this time slot.

2.2.5 Problem Statement

For a mobile device, it makes task scheduling decisions about offloading and local execution at the beginning of each slot. It then makes decisions for downloading the return data of cloud execution for the next slot at the end of current time slot. The objective of mobile devices is to keep all the queues stable and to minimize the overall energy consumption. The scheduling problem can be formulated as

$$\min: \limsup_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}\{\varepsilon(t) + p_O(t) + p_D(t)\} \tag{2.20}$$

$$\text{s.t. } \alpha^L(t) \cap \alpha^O(t) = \emptyset, \text{ for all } t \tag{2.21}$$

$$\left|\alpha^L(t)\right| \leq M, \text{ for all } t \tag{2.22}$$

$$\bar{Q} < \infty, \tag{2.23}$$

where Constraint (2.21) forbids a task to be both executed locally and offloaded to the cloud in the same time slot, Constraint (2.22) is the limitation forced by the number of cores in the CPU, and Constraint (2.23) ensures stability of the task and output queues. The optimal solution to the problem consists of cloud offloading or local execution decisions for each time slot t (i.e., $\alpha^L(t)$ and $\alpha^O(t)$) and the optimized CPU computation capability $\Theta(t)$ for each time slot t, which translates to the optimal CPU clock frequency f as discussed in Section 2.2.2 (configured as in (2.41)).

2.3 Task Scheduling Algorithm for Mobile Users

In this section, we present a task scheduling algorithm based on the Lyapunov optimization framework [9]. This algorithm requires no information about the stationary distributions of the arrival and wireless channel processes; it only requires information on the current queue lengths and the current channel conditions. Such an *online algorithm* property is useful for real-time applications [13, 14, 46].

2.3.1 Lyapunov Optimization Based Solution Algorithm

To present the proposed algorithm, we first define a Lyapunov function $L(Q(t))$ as in [9].

$$L(Q(t)) \triangleq \frac{1}{2}\sum_{i=1}^{N} Q_i^2(t) + \frac{1}{2}\sum_{i=1}^{N}\{Q_i^D(t)\}^2, \tag{2.24}$$

where $L(Q(0)) = 0$. If all the queue lengths are small, then $L(Q(t))$ will be small; if at least one queue is congested, then $L(Q(t))$ will become large. Since there is a finite number of applications running on the mobile device, $L(Q(t))$ being bounded is equivalent to the notion that the system is stable.

Since $L(Q(0)) = 0$, for $L(Q(t+1))$, we have

$$\mathbb{E}\{L(Q(t+1))\} = \mathbb{E}\left\{\sum_{k=0}^{t}[L(Q(k+1)) - L(Q(k))]\right\}$$
$$= \sum_{k=0}^{t}\mathbb{E}\{L(Q(k+1)) - L(Q(k))|Q(k)\} = \sum_{k=0}^{t}\Delta(L(k)),$$

where $\Delta(L(t))$ is the *drift* defined as [15]

$$\Delta(L(t)) \triangleq \mathbb{E}\{L(Q(t+1)) - L(Q(t))\,|Q(t)\}. \tag{2.25}$$

We can minimize $\Delta(L(t))$ to maintain a low expectation for $L(Q(t))$. It follows (2.16) that

$$Q_i^2(t+1) \leq \{Q_i(t) + A_i(t) - B_i(t) - B_i^O(t)\}^2. \tag{2.26}$$

For $i \in \alpha^O(t)$, we have

$$\{Q_i^D(t+1)\}^2 \leq \{Q_i^D(t) + B_i^O(t) - B_i^D(t)\}^2. \tag{2.27}$$

For $i \notin \alpha^O(t)$, we have

$$\{Q_i^D(t+1)\}^2 \leq \{Q_i^D(t) - B_i^D(t)\}^2. \tag{2.28}$$

Substituting (2.26), (2.27), and (2.28) into (2.25), we derive the drift (2.30) as follow.

$$\Delta(L(t)) \tag{2.29}$$
$$\leq \Phi + \mathbb{E}\left\{\sum_{i\notin\alpha^O(t)} Q_i(t)(A_i(t) - B_i(t)) - Q_i^D(t)B_i^D(t)\right\}$$
$$+\mathbb{E}\left\{\{Q_i(t)A_i(t) - (Q_i(t) - Q_i^D(t))B_i(t) - Q_i^D(t)B_i^D(t)\}|_{\{i\in\alpha^O(t)\}}\right\}$$

$$= \Phi + \mathbb{E}\{\varphi\}. \tag{2.30}$$

In (2.30), φ denotes the terms in the expectation operators and

$$\Phi = \frac{1}{2}\sum_{i=1}^{N}\mathbb{E}\left\{\{A_i(t) - B_i(t) - B_i^O(t)\}^2 + \{B_i^O(t) - B_i^D(t)\}^2\right\}. \tag{2.31}$$

Note that $B_i^O(t) = 0$ for $i \notin \alpha^O(t)$. If the arrival rate and service rate of each queue is bounded, which is true for stable systems, then Φ is bounded.

As in [9], we obtain the *drift-plus-penalty*, defined as $\Delta(L(t)) + V_p \cdot \mathbb{E}\{P(t)\}\}$, by scaling the energy consumption with a positive coefficient V_p. The parameter V_p indicates the user's emphasis on energy consumption. Following (2.30), the upper bound of the *drift-plus-penalty* can be derived as

$$\Delta(L(t)) + V_p \cdot \mathbb{E}\{P(t)\} \leq \Phi + \mathbb{E}\left\{\varphi + V_p \cdot P(t)\right\}. \tag{2.32}$$

To minimize the drift-plus-penalty, we can instead minimize $\{\varphi + V_p \cdot P(t)\}$ at every time slot, which only requires the current information on queue lengths, channel conditions, and the price for offloading.

Since there are M cores in the CPU of the mobile device, only M application can be executed by the CPU in each time slot. We assume that only one application can be offloaded at each time slot (through the single active wireless connection). We can derive the minimization expression as given in (2.33).

$$\begin{aligned} &\min\{\varphi + V_p P(t)\} \\ &= \min\left\{\sum_{i=1}^{N} Q_i(t)A_i(t) - \sum_{i=1}^{N} Q_i^D(t)B_i^D(t) - \sum_{i\notin\alpha^O(t)} Q_i(t)B_i(t)\right. \end{aligned} \tag{2.33}$$

$$
\begin{aligned}
&- (Q_i(t) - Q_i^D(t))B_i^O(t)|_{i\in\alpha^O(t)} + V_p P(t)\Bigg\} \\
= &\sum_{i=1}^{N} Q_i(t)A_i(t) + \min\Bigg\{V_p p_D(t) - \sum_{i=1}^{N} Q_i^D(t)B_i^D(t) + V_p\varepsilon(t) - \sum_{i\notin\alpha^O(t)} Q_i(t)B_i(t) \qquad (2.34)\\
&+ V_p p_O(t) - (Q_i(t) - Q_i^D(t))B_i^O(t)|_{i\in\alpha^O(t)}\Bigg\} \\
= &\sum_{i=1}^{N} Q_i(t)A_i(t) + \min\left\{V_p p_D(t) - Q_i^D(t)B_i^D(t)|_{i\in\alpha^D(t)}\right\} \qquad (2.35)\\
&+ \min\Bigg\{V_p\varepsilon(t) - \sum_{i\in\alpha^L(t)} Q_i(t)B_i(t) + \{V_p p_O(t) - (Q_i(t) - Q_i^D(t))B_i^O(t)|_{i\in\alpha^O(t)}\}\Bigg\} \\
= &\sum_{i=1}^{N} Q_i(t)A_i(t) + \min\{H_1\} + \min\{H_2\}. \qquad (2.36)
\end{aligned}
$$

The first term in (2.33), $\sum_{i=1}^{N} Q_i(t)A_i(t)$, only depends on the current queue lengths and arrival rates. It does not affect the offloading downloading decision for this time slot. We need to minimize the second term

$$H_1 = V_p p_D(t) - Q_i^D(t)B_i^D(t)|_{i\in\alpha^D(t)}, \qquad (2.37)$$

as a function of $\alpha^D(t)$, and the third term

$$
\begin{aligned}
H_2 = V_p\varepsilon(t) - &\sum_{i\in\alpha^L(t)} Q_i(t)B_i(t) + \\
&\{V_p p_O(t) - (Q_i(t) - Q_i^D(t))B_i^O(t)|_{i\in\alpha^O(t)}\}, \qquad (2.38)
\end{aligned}
$$

as a function of $\alpha^L(t)$, $\alpha^O(t)$, and $\Theta(t)$.

Notice that for $\min\{H_1\}$, with the expectation of power consumption and offloading data, we need to find a proper $\alpha^D(t)$ that minimizes $-Q_i^D(t)B_i^D(t)$ in order to minimize the following function.

$$\xi_i^D = V_p p_D(t) - Q_i^D(t)B_i^D(t). \qquad (2.39)$$

This can be done by evaluating (2.39) for every application in $\mathcal{N}'$ to find the application i having the smallest ξ_i^D. Recall that $B_i^D(t)$ is defined in (2.19). For a given downlink capacity $\omega_D(t)$, tasks with smaller data size and longer queue length tend to have a smaller $-Q_i^D(t)B_i^D(t)$. Note that $V_p p_D(t) - Q_i^D(t)B_i^D(t)|_{i\in\alpha^D(t)} = 0$ when $\alpha^D(t) =$ 'idle'. Thus a task will be offloaded in time slot t only when $\min\{V_p p_D(t) - Q_i^D(t)B_i^D(t)\} < 0$, meaning the channel condition is good or at least one of the task queues is long.

For the other term H_2, we need to minimize it by tuning $\alpha^L(t)$, $\alpha^O(t)$, and$\Theta(t)$. The term $V_p\varepsilon_i(t) - Q_i(t)B_i(t)$ can be rewritten as

$$\begin{aligned} & V_p\varepsilon_i(t) - Q_i(t)B_i(t) \\ &= V_p\eta\Theta_i^2(t) - Q_i(t)\cdot \arg\max_{\{b\}} \left\{ \sum_{k=1}^{b} \theta_i(k,t) \le \Theta_i(t) \right\} \\ &\cong V_p\eta\Theta_i^2(t) - Q_i(t)\frac{\Theta_i(t)}{\bar{\theta}_i(t)}, \end{aligned} \tag{2.40}$$

where $\bar{\theta}_i(t) = \frac{1}{Q_i(t)}\sum_{k=1}^{Q_i(t)}\theta_i(k,t)$. We can derive the approximate minimum value $V_p\varepsilon(t) - Q_i(t)B_i(t)$ subject to the CPU computation capability $\Theta_i(t)$ as

$$\xi_i^L(t) = -\frac{Q_i^2(t)}{4V_p\eta\bar{\theta}_i^2(t)}, \text{ when } \Theta_i(t) = \frac{Q_i(t)}{2V_p\eta\bar{\theta}_i(t)}. \tag{2.41}$$

Similarly, we can evaluate (2.41) for all the applications in $\mathcal{N}$ and find the minimizer. Since the computational capability of the CPU cannot be increased indefinitely, we set an upper bound for the CPU power, e.g., 10 W in this chapter.

For the term $\{V_p p_O(t) - (Q_i(t) - Q_i^D(t))B_i^O(t)|_{i\in\alpha^O(t)}\}$, we can minimize it by tuning $\alpha^O(t)$. Denoting

$$\xi_i^O = V_p p_O(t) - (Q_i(t) - Q_i^D(t))B_i^O(t) < 0, \tag{2.42}$$

an application $i \in \mathcal{N}'$ with smaller offloading data size and greater $Q_i(t) - Q_i^D(t)$ will achieve a smaller ξ_i^O. Also note that $\{V_p p_O(t) - (Q_i(t) - Q_i^D(t))B_i^O(t)|_{i\in\alpha^O(t)}\} = 0$ when $\alpha_i^O =' idle'$. Thus a task can be offloaded only when $\xi_i^O < 0$.

Then the $\min\{H_2\}$ term can be rewrite as

$$\min\{H_2\} = \min\left\{\sum_{i\in\alpha^L(t)} \xi_i^L + \xi_j^O|_{j\in\alpha^O(t),\alpha^O(t)\cap\alpha^L(t)=\emptyset}\right\}.$$

According to the above evaluation, the problem becomes

$$\begin{aligned}
&\sum_{i=1}^{N} Q_i(t)A_i(t) + \min\{H_1\} + \min\{H_2\} \\
=& \sum_{i=1}^{N} Q_i(t)A_i(t) + \min\left\{\xi_i^D\right\} + \\
&\min\left\{\sum_{i\in\alpha^L(t)} \xi_i^L + \xi_j^O|_{j\in\alpha^O(t),\alpha^O(t)\cap\alpha^L(t)=\emptyset}\right\},
\end{aligned} \tag{2.43}$$

where ξ_i^D, ξ_i^L, and ξ_j^O are defined in (2.39), (2.41) and (2.42), respectively. We also have $\alpha^O(t) \cap \alpha^L(t) = \emptyset$, since the same application cannot be executed locally and offloaded to the cloud in the same time slot. The proposed task scheduling algorithm is presented in Algorithm 1, where all computations except Step 2 are simple operations.

For Step 2 in Algorithm 1, the task scheduling can be illustrated as a minimum weighted matching of a bipartite graph as shown in Fig. 2.2. In the graph, vertex Application i, $i = 1, 2, \cdots, N$ represent the applications, vertex Core i, $i = 1, 2, \cdots, L$ stands for the cores in the CPU, and vertex OffLoad stands for the offloading link. The edge between vertice Application i and Core j means that it can be executed locally on core j and the weight of the edge is ξ_i^L. Correspondingly, the edge between vertice Application i and OffLoad means that it can be offloaded to cloud, while the weight of the edge is ξ_i^O. In Step 2, we need to find the selection edges with minimum weight, and according to constraint (2.21) and (2.22), each vertex can only be connected

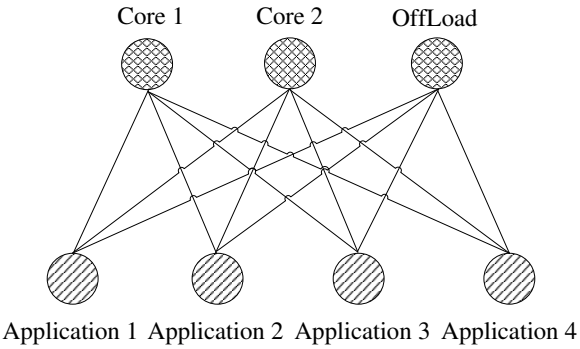

Figure 2.2: Task scheduling as a minimum weighted matching of a bipartite graph (illustrated for $N = 4$ and $M = 2$).

Algorithm 1: Task Scheduling Algorithm

1 Update all the task queues and estimate wireless link capacities at the beginning of time slot t ;
2 Find the minimum combination of $\sum_{i \in \alpha^L(t)} \xi_i^L + \xi_j^O$, where $\alpha^O(t) \cap \alpha^L(t) = \emptyset$ and $j \in \mathcal{N}'$;
3 **if** $\xi_j^O < 0$ **then**
4 | Offload tasks of application j to the cloud ;
5 **end**
6 **for** $i \in \alpha^L(t)$ **do**
7 | **if** $\xi_i^L < 0$ **then**
8 | | Execute tasks of application i locally, with CPU capacity $\Theta_i(t) = \frac{Q_i(t)}{2V_p \eta \theta_i(t)}$;
9 | **end**
10 **end**
11 Find the minimum ξ_i^D;
12 **if** $\xi_i^D < 0$ **then**
13 | Fetch the output data for application i tasks from the cloud ;
14 **end**

with one selected edge. Then it is a maximum weighted bipartite matching problem and can be solved with Hungarian algorithm [16] with complexity $O(N * (M+1)^2)$ if $(M + 1 < N)$, or $O((M + 1) * N^2)$ otherwise.

In Algorithm 1, at the beginning of each time slot t, the mobile device first update the queues of tasks and estimate the capacity of wireless capacities to compute ξ_i^L, ξ_i^O, and ξ_i^D. In Step 2, it find out smallest combination of $\sum_{i \in \alpha^L(t)} \xi_i^L + \xi_j^O$, where $\alpha^O(t) \cap \alpha^L(t) = \emptyset$, since a task should not be computed locally and offloaded to cloud at the same time. Then it offloads the corresponding task of application j if $\xi_j^O < 0$ and computes the tasks of application $i \in \alpha^L(t)$ if $\xi_i^L < 0$, with $\Theta_i(t) = \frac{Q_i(t)}{2V_p \eta \Theta_i(t)}$. At last, the mobile user make the decision of downloading the output of cloud

computing. It first find the smallest ξ_i^D for all applications in $\mathcal{N}'$. If $\xi_i^D < 0$ for the smallest ξ_i^D, then it download the corresponding output of cloud computing.

2.3.2 Performance Analysis

Following the framework of Lyapunov optimization [9], we derive the upper bounds for the expected average power consumption and the expected average queue length achieved by the proposed algorithm, which are summarized in the following theorem. The proof is presented in the Appendix.

Theorem 2.1. *Assume that the arrival rate of tasks $\vec{\lambda}$ is strictly within the system capacity region. That is, the system can maintain stability under certain $\{\alpha^L(t), \alpha^O(t), \alpha^D(t), \Theta(t)\}$. Then the bounds on average energy consumption and queue length under Algorithm 1 can be written as*

$$\limsup_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T-1} E\{P(t)\} \leq P^{opt} + \frac{\Phi}{V_p} \tag{2.44}$$

$$\limsup_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T-1} \sum_{i=1}^{N} E\{Q_i(t) + Q_i^D(t)\} \leq \frac{1}{\epsilon}(\Phi + V_p P), \tag{2.45}$$

where P^{opt} is the minimum energy consumption a stable system can achieve, P is the average energy consumption under the proposed algorithm, and $\epsilon > 0$ is the distance between the data arrival rate vector $\vec{\lambda}$ and the system capacity region under the proposed algorithm.

Theorem 2.1 demonstrates the trade-off between energy consumption and queue length (or, delay). The upper bound of the average energy consumption is $O(1/V_p)$ and the upper bound of the average queue length is $O(V_p)$. Therefore these are conflicting objectives. We can tune V_p to flexibly trade off between energy consumption and queue length. When the power supply is not so limited (e.g., a charger is available), the user can increase V_p to reduce the queue length (and thus delay) and enjoy better quality of experience (QoE). On the other hand, if the power constraint is stringent (e.g., the mobile device is running out of battery and no charger is available), the user can decrease V_p to save energy at the expense of longer average queue length and larger delay.

2.4 Trace-driven Simulation Validation

We evaluate the performance of the proposed algorithm with trace-driven simulations. In the simulations, we adopt the wireless network measurement data gathered by testing the data rate of the LTE/WiFi networks while walking around the Auburn University campus with an iPhone5. The LTE carrier is AT&T and the WiFi network is deployed by Auburn university. In particular, half of the LTE rate tests are conducted outdoor and half of the tests are conducted indoor. The WiFi rate tests are conducted in Broun Hall in the Auburn University Campus.

In the simulations, the wireless link data rate is randomly selected from the measured trace. For power consumption, we adopt the power models for LTE and WiFi proposed in [17]. For the uplink, the LTE power model can be approximated as $p_O = a_{LTE} \cdot \omega_O + b_{LTE}$, where a_{LTE} = 0.5 W, b_{LTE} = 1.25 W, and ω_O is the wireless network data rate in Mbps. For WiFi, the power consumption mode is $p = a_{WiFi} \cdot \omega_O + b_{WiFi}$, where a_{WiFi} = 0.24 W and b_{WiFi} = 0.125 W. For downlink, the power model for LTE can be approximated as $p_D = a^D_{LTE} \cdot \omega_D + b^D_{LTE}$, where a^D_{LTE} = 0.042 W, b^D_{LTE} = 1.25 W. For WiFi, the power consumption mode is $p_D = a^D_{WiFi} \cdot \omega_D + b^D_{WiFi}$, where a^D_{WiFi} = 0.12 W and b^D_{WiFi} = 0.125 W.

We consider a scenario with five applications running in the mobile device and all of them can be offload. The task arrival rate of each application ranges from 0.5 to 2.0. The offloading data size of the tasks follows a truncated Exponential distribution with means ranging from 60 KB to 300 KB. For local execution, η was set to 0.6 corresponding to the normalized computation complexity Θ. The normalized computation complexity of each task follows an Exponential distribution with means ranging from 0.1 to 1. In the simulations, V_p is increased from 1 to 200. For each V_p value, the simulation runs for 50,000 time slots.

We compare the following four schemes in the simulations: (i) the proposed scheme with single core CPU, (ii) the proposed scheme with dual core CPU, (iii) the proposed scheme with single core CPU, and with Large Output of Cloud Computing (LOCC) (i.e. the average data size

of cloud computing is twice of that of offloading), and (iv) the "eTime" strategy proposed in [15] with LOCC.

The simulation results are plotted in Figs. 2.3 and 2.4 for average queue length and average power consumption, respectively. It can be seen that there is a clear trade-off between average energy consumption and average queue length achieved by tuning V_p for both single core and dual core CPU. When V_p is increased, the average energy consumption is decreased but the average queue length is increased. It confirms the findings in Theorem 2.1 that the average queue length follows $O(V_p)$ (see Fig. 2.3) and the average energy consumption follows $O(1/V_p)$ (see Fig. 2.4) asymptotically. When V_p is smaller than 10, the energy consumption decreases rapidly with V_p, while the average queue length increases almost linearly with V_p. Therefore, users can achieve big energy savings, while only suffers a linearly increased delay, by increasing V_p in this region. From the simulation, we can find clearly that for dual core CPU, the queue length is much shorter than that of the single CPU system. But the power consumption for dual core is much higher with small V_p, but with high V_p (i.e., larger than 4), the system with dual core CPU enjoy lower energy consumption. It means that system with dual core system enhances the system computation ability and show greater flexibility for trade off between energy consumption and queue length.

For system with single core CPU with LOCC, it suffers from longer queue length and greater energy consumption with large V_p (i.e. greater than 4), as the downloading for Output of Cloud Computing is more resource consumption. The queue length of the single core CPU system with LOCC suffers from a high queue length with the low V_p(i.e. smaller than 4), that is because the system offloading tasks aggressively with low V_p and the downloading for output of cloud is resource consuming, which increases the average queue length. With low V_p (i.e. smaller than 4), the power consumption of single core CPU system with LOCC consumes less energy consumption than that of single core CPU system. It is because that the single core CPU system with LOCC has longer queue for downloading the output of cloud computing, which result in a smaller ξ_j^O and effects of V_p is enhanced.

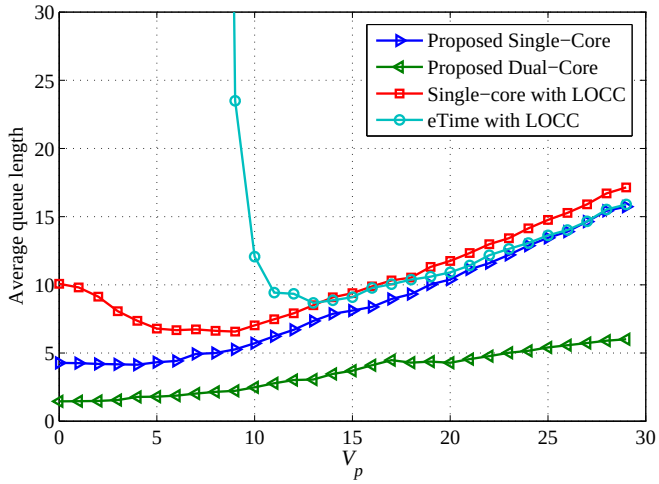

Figure 2.3: Average queue length of the four schemes.

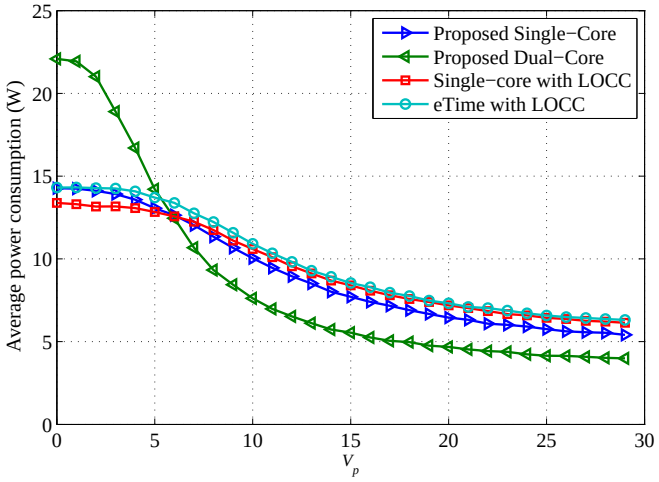

Figure 2.4: Average power consumption of the four schemes.

The simulation results also demonstrate that the performance of the proposed algorithm is better than that of the strategy proposed in [15] with LOCC, which suffers higher energy consumption. In addition, in the LOCC scenario, eTime couldn't stabilize the system with a low V_p. It is because that with a low V_p, eTime aggressively offloads tasks to the cloud but couldn't download the output of cloud execution.

2.5 Related Work

Cloud offloading is regarded as an effective solution to save energy, extend storage spaces, and enable computation intensive applications at mobile devices [4–6]. There have been many prior work addressing the various design issues of cloud computing to fully harvest its potential [7, 18–21, 24–26]. In particular, considerable recent works have focused on building the framework of enable mobile computation offloading [7, 21, 24–26], suggesting for a mobile device to execute codes remotely in resource-rich servers, which connect the mobile device through LAN or wireless link. Ref. [25] implemented method level offloading for applications on Microsoft .NET, and Ref. [26] implemented a flexible application partitioner which enables seamlessly offloading of part of the execution to the virtual machine. On the other hand, many other works [18, 27, 30] have focused on backing up data and applications to extend the storage space of mobile devices. However, both computation offloading and data/application backup involve considerable energy consumption for data transmission between mobile devices and the cloud, which may makes some excellent techniques [32] infeasible in practical implementation scenarios.

Researchers have started to investigate the energy cost of offloading [11, 15, 19, 22, 23, 25, 28–31, 33, 35]. Some techniques focused on reducing the energy consumption during offloading [22, 25, 29–31, 33, 33]. For example, in [22], the authors proposed a dynamic offloading algorithm to save energy by offloading some components of an application to the cloud, while Ref. [33] proposed an algorithm to reduce energy consumption by selecting the most energy efficient WiFi AP for offloading. Furthermore, some researches have investigated the tradeoff between energy consumption and delay [11, 19, 23, 28, 34]. For example, the bandwidth and energy costs of cloud computing were investigated in [11]. In [28], a heuristic algorithm was proposed to jointly minimize the energy consumption and delay. However, these works are based on static models of application, and more important, the stochastic characteristics of applications and network dynamics have not been taken into consideration. The authors of [19, 23] proposed an energy-optimal mobile computing framework under stochastic wireless channels, while considering the single application

scenario. In [15], an energy-efficient transmission algorithm between the cloud and mobile devices was presented based on the Lyapunov optimizing framework [9]. However, the local computation resources in the mobile devices has not been fully utilized, and it doesn't consider downloading the cloud execution output.

This work was motivated by the above interesting works to investigate the energy-delay trade-off in cloud offloading with a Lyapunov optimization approach. We explicitly considered the stochastic nature of both user and application behaviors, and network dynamics, and addressed the more challenging case of multiple applications, thus greatly extending the work in [19, 23]. This work also extended prior work [15] by considering multi-core CPUs and fully utilizing the local computing capability, by making offloading decisions based on both task queues and queues for downloading the output of cloud execution. As in [15], the online operation of the proposed scheme makes it highly suitable for real-time applications.

2.6 Conclusions

In this chapter, we proposed a scheduling scheme for energy-efficient cloud offloading for muti-core mobile devices, while considering downloading the cloud execution output in the model. Based on Lyapunov optimization, we developed an online algorithm that does not require information about stationary distribution of applications and the network condition, making it amenable to real-time implementation for practical scenarios. We proved theoretical bounds for the proposed algorithm and validated its performance with trace-driven simulations.

2.7 Appendix

2.7.1 Proof Of Theorem 2.1

According to (2.30) and (2.33), we have

$$\min\{\varphi + V_p P(t)\} \tag{2.46}$$

$$
\begin{aligned}
&= \min\Bigg\{ V_p P(t) + \sum_{i=1}^{N} Q_i(t) A_i(t) - \sum_{i=1}^{N} Q_i^D(t) B_i^D(t) - \\
&\quad \sum_{i \notin \alpha^O(t)} Q_i(t) B_i(t) - (Q_i(t) - Q_i^D(t)) B_i^O(t)|_{i \in \alpha^O(t)} \Bigg\} \\
&= \min\Bigg\{ V_p P(t) + \sum_{i=1}^{N} Q_i^D(t)(B_i^O(t) - B_i^D(t)) + \\
&\quad \sum_{i=1}^{N} Q_i(t)(A_i(t) - B_i(t) - B_i^O(t)) \Bigg\} \\
&\leq V_p P^*(t) + \sum_{i=1}^{N} Q_i^D(t)(B_i^{*O}(t) - B_i^{*D}(t)) + \\
&\quad \sum_{i=1}^{N} Q_i(t)(A_i(t) - B_i^*(t) - B_i^{*O}(t)),
\end{aligned}
$$

where $P^*(t)$, $B_i^*(t)$, $B_i^{*O}(t)$ and $B_i^{*D}(t)$ are the terms corresponding to any other (possibly randomized) feasible schedule. Now consider a randomized scheduling policy that achieves the following for Application $i \in \mathcal{N}$.

$$\mathbb{E}\{P^*(t)\} = P^{opt} \tag{2.47}$$

$$\mathbb{E}\left\{B_i^{*O}(t) - B_i^{*D}(t)\right\} \leq 0 \tag{2.48}$$

$$\mathbb{E}\left\{A_i(t) - B_i^*(t) - B_i^{*O}(t)\right\} \leq 0, \tag{2.49}$$

where P^{opt} is the minimum power consumption a stable system can achieve and (2.48) and (2.49) stabilize the queues.

For the proposed algorithm, we have

$$
\begin{aligned}
&\Delta(L(t)) + V_p \cdot \mathbb{E}\{P(t)\} \qquad (2.50) \\
&\leq V_p \cdot \mathbb{E}\{P(t)\} + \Phi + \mathbb{E}\{\varphi\} \\
&\leq V_p \cdot \mathbb{E}\{P^*(t)\} + \mathbb{E}\left\{ \sum_{i=1}^{N} Q_i^D(t)(B_i^{*O}(t) - B_i^{*D}(t)) \right\} +
\end{aligned}
$$

$$\mathbb{E}\left\{\sum_{i=1}^{N} Q_i(t)(A_i(t) - B_i^*(t) - B_i^{*O}(t))\right\} + \Phi$$
$$\leq V_p \cdot P^{opt} + 0 + \Phi,$$

where

$$\mathbb{E}\left\{\sum_{i=1}^{N} Q_i^D(t)(B_i^{*O}(t) - B_i^{*D}(t))\right\} \leq 0 \tag{2.51}$$
$$\mathbb{E}\left\{\sum_{i=1}^{N} Q_i(t)(A_i(t) - B_i^*(t) - B_i^{*O}(t))\right\} \leq 0, \tag{2.52}$$

according to (2.48) and (2.49).

Then we have

$$\Delta(L(t)) + V_p \cdot \mathbb{E}\{P(t)\} \leq V_p \cdot P^{opt} + \Phi, \tag{2.53}$$

and $\sum_{k=0}^{T-1} \Delta(L(t)) = L(T) < \infty$ for a stable system. It follows that

$$\limsup_{t\to\infty} \frac{1}{T} \sum_{k=0}^{T-1} \Delta(L(T)) + \limsup_{t\to\infty} \frac{V_p}{T} \sum_{k=0}^{T} \mathbb{E}\{P(t)\}$$
$$= 0 + \limsup_{t\to\infty} \frac{V_p}{T} \sum_{k=0}^{T} \mathbb{E}\{P(t)\}$$
$$\leq V_p \cdot P^{opt} + \Phi.$$

Then we have that (2.44) holds true.

Suppose for Application $i \in \mathcal{N}$, there exist some real number $\epsilon > 0$, such that

$$\mathbb{E}\left\{B_i^O(t) - B_i^D(t)\right\} \leq -\epsilon \tag{2.54}$$
$$\mathbb{E}\left\{A_i(t) - B_i(t) - B_i^O(t)\right\} \leq -\epsilon. \tag{2.55}$$

According to (2.46), we have

$$\begin{aligned}
&\Delta(L(t)) + V_p \cdot \mathbb{E}\{P(t)\} \qquad (2.56)\\
\leq\ & V_p \cdot \mathbb{E}\{P(t)\} + \mathbb{E}\left\{\sum_{i=1}^{N} Q_i^D(t)(B_i^O(t) - B_i^D(t))\right\} + \\
& \mathbb{E}\left\{\sum_{i=1}^{N} Q_i(t)(A_i(t) - B_i(t) - B_i^O(t))\right\} + \Phi \\
\leq\ & V_p \cdot \mathbb{E}\{P(t)\} + \Phi - \epsilon \cdot \mathbb{E}\left\{\sum_{i=1}^{N}(Q_i(t) + Q_i^D(t))\right\}.
\end{aligned}$$

As $\Delta(L(t)) + V_p \cdot \mathbb{E}\{P(t)\} \geq 0$, we have

$$\mathbb{E}\left\{\sum_{i=1}^{N}(Q_i(t) + Q_i^D(t))\right\} \leq \frac{1}{\epsilon}\{V_p \cdot \mathbb{E}\{P(t)\} + \Phi\}. \qquad (2.57)$$

It follows that

$$\begin{aligned}
&\limsup_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T-1}\sum_{i=1}^{N} E\{Q_i(t) + Q_i^D(t)\} \qquad (2.58)\\
\leq\ & \frac{\Phi}{\epsilon} + \frac{1}{\epsilon}\limsup_{T\to\infty}\frac{1}{T}\{V_p\mathbb{E}\{P(t)\}\} \\
=\ & \frac{1}{\epsilon}(\Phi + V_p P),
\end{aligned}$$

and we conclude that (2.45) holds true.

Chapter 3

Inter-operator Opportunistic Spectrum Sharing in LTE-unlicensed

3.1 Introduction

With the unprecedented growth in wireless data, wireless operators are in critical need of more spectrum for higher capacity. To meet the so-called 1000x mobile data challenge [87], extending LTE to the unlicensed spectrum, as specified in LTE Rel-10 – Rel-13 [83, 84], has recently gained significant attention [83, 84, 87, 88, 90, 92–98]. However, there are two main challenges to the success of the so-called *LTE-unlicensed* technology. First, the unlicensed bands are already occupied by many existing wireless networks (e.g., WiFi). It is essential to enable the coexistence of LTE-unlicensed with existing unlicensed band users, i.e., to avoid significant performance degradation to existing users while achieving high capacity gains with LTE-unlicensed. Second, the interference in unlicensed bands is unpredictable, which is detrimental to the performance of LTE-unlicensed users. Hence, it is important to effectively manage the interference between LTE-unlicensed and existing users, and that among LTE-unlicensed users themselves.

To study the coexistence of LTE-unlicensed with existing unlicensed band users, some system level simulation studies have been reported in several recent works [88, 93, 94]. The simulation results show that the WiFi performance could be significantly degraded, while the LTE performance is only slightly affected. This is because WiFi uses Carrier Sensing Multiple Access (CSMA) to compete for channel access, while LTE adopts a centralized channel access control mechanism. WiFi usually keeps silent when sensing a busy channel continuously used by LTE. To protect existing unlicensed band users, requirements for clear channel assessment (CCA) and Listen Before Talk (LBT) are specified by European standardization bodies [95]. In LBT, a user equipment (UE)

must perform CCA on the operating channel(s) before starting a transmission. The observing duration should be at least 20 μs.

Although the LTE performance may be only slightly affected by WiFi in some coexistence scenarios [93, 94], there could still be significant throughput degradations due to the inter-operator interference, when multiple LTE-unlicensed base stations (BS) of different operators are deployed in the same area [83]. There are two solutions to this problem: (i) make an agreement for the operators to allocate the unlicensed spectrum; or (ii) enable opportunistic access to unlicensed channels. The first solution may not be practical in most countries due to competition among operators and the lack of regulation for unlicensed bands [83], while the second solution is promising for effective unlicensed spectrum sharing.

In this work, we investigate the problem of opportunistic spectrum sharing among LTU-unlicensed BS's. We consider the License Assisted Access (LAA) scenario, in which licensed and unlicensed carrier bands are integrated and used [84]. We also adopt the LBT mechanism for co-existence of LTE-unlicensed and WiFi [95]. For the LTE-unlicensed BS's deployed in the same area on both licensed and unlicensed bands, we propose a novel distributed online algorithm for opportunistic sharing of unlicensed bands among the BS's, while guaranteeing the QoS of UEs in the form of bounded worst case delay and minimized packet drop rate.

Specifically, based on Lyapunov optimization, we first derive an online algorithm for BS's to evaluate the true value of unlicensed spectrum, guarantee a maximum delay, and minimize the packet drop rate. We then develop a distributed auction mechanism to incorporate the Lyapunov optimization based schemes, aiming to maximize the social welfare in each auction and enable optimal spectrum reuse. We prove that all the BS's bid truthfully with the proposed algorithm, while the UEs' QoS requirements on delay and packet drop rate can be guaranteed with bounded optimality gaps. The proposed algorithms are validated with simulations and are shown to outperform two benchmark schemes with considerable gains in all the cases simulated in this work.

This work presents a comprehensive and effective solution to the problem of opportunistic spectrum sharing for LTE-unlicensed. The algorithm design is based on rigorous theoretic model

and analysis. Due to the Lyapunov optimization approach, the proposed algorithms are applicable to very general scenarios with different traffic models and service rate distributions. The proposed schemes are also *online* algorithms, i.e., only requiring the current state of the network (e.g., queue backlogs and channel conditions), making them highly suitable for practical implementations. In addition to proving several nice properties of the proposed algorithms, including truthful bidding, utility maximization, social welfare maximization, and packet drop rate minimization, we also reveal an interesting trade-off between delay and packet drop rate, which provides a useful control knob for operators.

The remainder of this work is organized as follows. We discuss related works in Section 3.2 and introduce the system model in Section 3.3. We discuss evaluation of unlicensed spectrum, resource allocation, and drop scheduling in Section 3.4. We present the proposed auction mechanism and analyze its performance in Section 3.5. Our simulation results are analyzed in Section 3.6. Section 3.7 concludes this work.

3.2 Related Work

The considerable amount of underutilized spectrum in unlicensed bands is the main motivation for operators and researchers to extend LTE, a well-designed OFDMA solution, to unlicensed bands [82–84,87,88,90,92–98]. One of the biggest challenges is the coexistence of LTE-unlicensed and WiFi [83, 87, 88, 90, 92–97, 102–104]. In [93, 94], system level simulations were conducted to evaluate the feasibility of LTE/WiFi coexistence. It was shown that such coexistence causes significant degradations to the WiFi performance. but only affects the LTE performance slightly. Hence, LBT was introduced to protect the WiFi users in the coexistence scenario [95, 104], where an LTE-unlicensed BS follows a CCA process before accessing the unlicensed spectrum. In [92], an analytical model was presented for evaluating the effectiveness of the simple LBT. The analysis showed that LBT can effectively mitigate the impact of LTE-unlicensed on WiFi, though the performance of LTE-unlicensed would be degraded. Furthermore, experiments [82], show that with

LBT or adaptive duty cycle, WiFi can be will protected. Therefore, we consider LBT in this work to address the coexistence issue of LTE-unlicensed and WiFi.

Another challenge in LTE-unlicensed is interference management among LTE-unlicensed BS's [83], while opportunistic spectrum sharing is one of the proposed solutions. In [99], a credit token based spectrum auction scheme was proposed for spectrum leasing among secondary users, while in [100], a revenue generation for truthful spectrum auction in dynamic spectrum access was proposed to render a truthful bidding for spectrum leasing from agencies. In a recent work [101], a socially-optimal online spectrum auction is proposed for spectrum sharing among secondary users. However, these works either fail to address the new challenges for spectrum sharing in LTE-unlicensed, or provide no precise evaluation of the value of spectrum based on QoS guarantees in auctions. In [98], a game theoretic approach is proposed to enable spectrum sharing among LTE-unlicensed BS's through power control. However, it neglects to exploit the potential advantage of spectrum reuse among the BS's.

Motivated by the interesting prior work and the high potential of LTE-unlicensed, we propose a distributed online auction scheme for LTE-unlicensed BS's. The goal is to maximize the expected social welfare in each auction through efficient assignment and spectrum reuse, as well as meeting the QoS requirement of maximum delay and minimizing the packet drop rate at the same time.

3.3 System Model

3.3.1 LTE-unlicensed Network Model

We consider the LAA scenario, in which licensed and unlicensed carrier bands are integrated and used [84]. This can be enabled by Carrier Aggregation (CA) defined in LTE Rel-10 – Rel-13 [83, 84]. With LAA, LTE on licensed band serves as a backbone and the CA of unlicensed bands boosts the downlink (FDD) or both downlink and uplink (TDD) capacity [87]. Considering the asymmetric uplink and downlink traffic, we focus on the downlink transmission of LAA in the FDD scenario, in which the unlicensed carrier bands are utilized to enhance downklink data

transmission. Due to the low power constraint on unlicensed spectrum imposed by regulations (e.g., WiFi standards) and the relatively higher frequency of unlicensed bands (i.e., 5GHz), it is expected to have coverage holes in unlicensed band with co-site deployment of licensed and unlicensed bands. Hence, we consider non-co-site deployment of licensed and unlicensed bands in this work.

Specifically, we consider a system with M BS's operating in the LTE-unlicensed mode, denoted as $\mathcal{M} = \{1, 2, \ldots, M\}$. The BS's could be several Macro eNBs of different operators operating on both licensed and unlicensed bands, and/or pico nodes working on unlicensed bands. We also assume a high speed backhaul for coordinating the operation of the BS's, e.g., inter-cell interference coordination (ICIC) and bidding information exchange as in our proposed scheme. Define the interference index variable for BS i and j as[1]

$$I_{i,j} = \begin{cases} 1, & \text{if BS } i \text{ and } j \text{ interfere with each other} \\ 0, & \text{otherwise.} \end{cases} \tag{3.1}$$

Let $\mathcal{U}^m = \{1, 2, \ldots, U^m\}$ denote the set of UEs served by LTE-unlicensed BS m, which maintains a queue for each UE i, denoted as Q_i^m. Let $\mathcal{C} = \{1, 2, \ldots, C\}$ be the set of orthogonal channels, each of which has an identical bandwidth as the corresponding WiFi channel. Furthermore, there is no overlap between two different channels and none of the channels overlaps with more than one WiFi channels (i.e., they are "aligned"). We adopt the LBT mechanism for LTE-unlicensed/WiFi co-existence [95]. Moreover, any transmission of an LTE-unlicensed BS must be followed by an idle period of the channel to avoid starvation of WiFi users. The transmission time of LTE-unlicensed BS's should be confined to one frame to limit the impact on coexisting WiFi users.

[1]We adopt the physical model in [107] to define the interference range of nodes.

3.3.2 Transmission And Qeueing Model

In this work, we consider the UEs covered by LTE with both licensed and unlicensed bands.[2] LTE on licensed bands provides relatively reliable data transmissions. We assume that for UE i, BS m provides a data rate on licensed bands that transmits $R_i^m(t)$ packets in frame t. With the LBT mechanism, an LTE-unlicensed BS needs to wait for an available frame on unlicensed bands and bid for transmission opportunity on the frame to avoid collision among the BS's.[3] If BS m wins the transmission opportunity on an unlicensed channel $c \in \mathcal{C}$ in frame t, then it can provide an extra data rate for UE $i \in \mathcal{U}^m$, denoted as $R_{ic}^m(t)$. We also have $R_{ic}^m(t) = \varphi_{ic}^m(t) e_{ic}^m(t)$, where $\varphi_{ic}^m(t)$ is the number of Resource Blocks (RBs) assigned to UE i, and $e_{ic}^m(t)$ is the expected data rate provided by an RB in packets per frame, which depends on the condition of channel c between BS m and UE i.[4]

For each UE i, $A_i^m(t)$ data packets arrive at BS m during frame t. We assume the arriving packets follow a certain process with a bounded maximum rate, i.e., $A_i^m(t) \leq (A_i^m)^{max}$. The queue at BS m for UE i is maintained as

$$\begin{aligned} & Q_i^m(t+1) \\ = \ & \max\{Q_i^m(t) - R_{ic}^m(t) - R_i^m(t) - d_i^m(t), 0\} + A_i^m(t), \end{aligned} \tag{3.2}$$

where $Q_i^m(0) = 0$ and $d_i^m(t)$ is the number of packets dropped at frame t due to violating the maximum delay requirement.

3.3.3 Spectrum Auction And LBT On Unlicensed Band

The success of LTE on unlicensed bands hinges upon the coexistence of LTE-unlicensed with other wireless networks on the same bands. LBT is introduced to enable the coexistence of

[2]For UEs with no coverage of LTE licensed band, LTE-unlicensed is not available due to the absence of a control channel. For UEs with no coverage of LTE-unlicensed band, the regular LTE service can be offered.

[3]On unlicensed spectrum, planning is not feasible since any operator can deploy a BS if it is desired to do so.

[4]We assume negligible frequency selective fading in each of the channels.

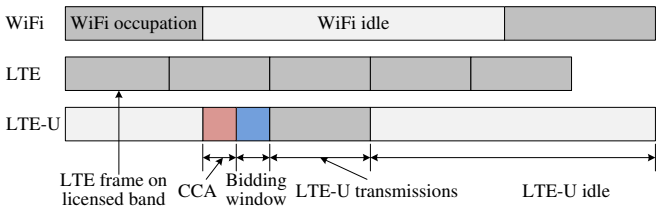

Figure 3.1: The frame structure of the proposed auction scheme, where LTE-unlicensed and WiFi share the same unlicensed channels.

LTE-unlicensed and WiFi. Before an LTE-unlicensed transmission, the BS should follow a CCA procedure and wait for an idle frame before claiming the channel to transmit. The CCA process of LBT can effectively prevent collision between LTE-unlicensed and WiFi. However, if more than one LTE-unlicensed BS's, within an interference range, claim and transmit on the same idle channel, there will still be collision among themselves. A channel bidding mechanism among LTE-unlicensed BS's is thus needed right after LBT.

Spectrum auction takes place among the LTE-unlicensed BS's that are interested in transmitting on an idle channel. After CCA, if a BS identifies an idle channel $c \in \mathcal{C}$, it may bid for the transmission opportunity. Other BS's can bid for the same channel following the first bid in the bidding window. All bids should be submitted to auction session initiated by the first bidder, denoted as the *auction initiator*, in its interference range. If there is no BSs in an active auction session for channel c in the interference range of a BS, the BS itself will become the auction initiator.[5] The auction is denoted as $S_c^{m^*}(t)$, where m^* is auction initiator, t is the frame that the winner BS/BS's access, c is the channel for auction, and $\{i \in S_c^{m^*}(t)\}$ are the BS's that participate in the auction. The frame structure of the auction is shown in Fig. 3.1. The auction can be conducted in the following *three* steps.

Step 1: Any BS $m \in \mathcal{M}$ interested in transmitting on channel c evaluates the value of transmitting on channel c for frame t, denoted as $\tilde{b}_c^m(t)$. It then submits a bid $b_c^m(t)$ to the auction session for transmission on the next frame.[6] Note that each BS aims to maximize its own utility in the

[5]The auction initiator serves as a virtual holder. The actual auction is processed in a back-end server to reduce the cost on the auction initiator and avoid cheating from it.

[6]If there are more than one auction sessions in its interference range, the BS will look for transmission opportunities on other channels. Such information can be obtained by sensing and/or information from a Geographic Information

auction, so it may try to manipulate the auction by submitting a bid deviating from its true value, i.e., $\tilde{b}_c^m(t) \neq b_c^m(t)$. In this work, we aim to design a strategy-proof auction to force BS's to bid truthfully (see Section 3.5.2).

Step 2: At the end of each bidding window, the auction session makes the channel assignment decision $\alpha_c^{m^*}(t)$, i.e., the set of auction winners to access channel c in the following transmission frames. Notice that the set of auction winners should be beyond the interference range of each other (i.e., $I_{i,j} = 0$, for all $i, j \in \alpha_c^{m^*}(t)$). The auction session decides the payment $\hat{b}^m(t)$ of all the BS's participating in the auction. Auction losers do not need to make a positive payment.

Step 3: At the beginning of transmission frame t, the winner BS's make decisions on transmission or dropping packets.

3.3.4 Utility Function And Social Welfare

We consider selfish BS's, each aiming to maximize its utility during each bidding cycle. The utility of BS $m \in \mathcal{M}$ depends on the QoS of the UEs it serves, including the drop rate and packet delay. The BS decides to bid when there is a potential transmission opportunity on channel c starting at frame t. If BS m participates in an auction of channel c that is available at frame t, its utility function is defined as

$$\phi_c^m(t) = \sum_{i \in \mathcal{U}^m} \{-\beta_i^m d_i^m(t)\} - \hat{b}_c^m(t), \tag{3.3}$$

where β_i^m is the penalty of dropping a packet of UE i served by BS m. Note that we do not include the delay constraint in the utility function, which, however, will be considered in the design of a dropping policy in next section. The transmission on licensed band is not included in the utility function because we aim to limit the modification on the current LTE system; and assume that the transmission on licensed band is not affected by the transmission on unlicensed band. However,

System (GIS), to the auction server to compete for the channel. If there are more than one channels available, then the BS can randomly choose *one* to bid.

the transmissions on licensed band do have a great influence on the queue length and packet drop rates of the UEs, which will be considered in the algorithm design.

The objective of the auction design is to maximize the social welfare of each auction. The social welfare of an auction on transmission opportunity at frame t on channel c should be the total utility of all anticipating BS's in auction $S_c^{m^*}(t)$. As payments are made among the participants, so the total payment should always be 0. Hence, the social welfare of auction $S_c^{m^*}(t)$ is defined as follows.

$$\sum_{m \in S_c^{m^*}(t)} \phi_c^m(t) = \sum_{m \in S_c^{m^*}(t)} \sum_{i \in \mathcal{U}^m} \left\{ -\beta_i^m d_i^m(t) \right\}. \tag{3.4}$$

3.4 Lyapunov Optimization based Valuation and Scheduling

3.4.1 Virtual Queue And Delay Bound

In each auction, a BS needs to dynamically evaluate the value of spectrum resource in LTE-unlicensed, and decide the resource allocation and packet drop scheme according to the channel condition and the queue length of each UE it serves. In this section, we apply Lyapunov optimization to derive an online algorithm for resource allocation and packet drop control to guarantee the maximum delay of packets [46, 86, 105]. For bidding on LTE-unlicensed bands, a successful bid would provide additional transmission opportunity for the next frame.

We adopt the ϵ-persistence queue [86] to guarantee the maximum delay requirement. The BS maintains the following *virtual queue* for each UE it serves.

$$\begin{aligned} Z_i^m(t+1) = \max \big\{ & Z_i^m(t) + \epsilon_i^m \cdot 1_{\{Q_i^m(t)>0\}} - R_i^m(t) - \\ & R_{ic}^m(t) - d_i^m(t) - Z_i^m(t) \cdot 1_{\{Q_i^m(t)=0\}}, 0 \big\}, \end{aligned} \tag{3.5}$$

where $\epsilon_i^m > 0$ is a prescribed constant; $1_{\{\cdot\}}$ is an indicator function; and $Z_i^m(0) = 0$. When $Q_i^m(t) > 0$, the virtual queue $Z_i^m(t)$ has the same departure process $R_{ic}^m(t) + R_i^m(t) + d_i^m(t)$ as

$Q_i^m(t)$, but its arrival rate is a constant ϵ_i^m. When $Q_i^m(t) = 0$, $Z_i^m(t)$ will be reset to 0. In fact, $Z_i^m(t)$ approximately tracks the packet delay of queue Q_i^m. A larger $Z_i^m(t)$ indicates a longer delay of packets in the real queue $Q_i^m(t)$. An algorithm that stabilizes $Z_i^m(t)$ and $Q_i^m(t)$ will ensure a bounded maximum delay, as given in the following Fact [86].

Fact 1. *(Upper Bound of Delay) Suppose $Q_i^m(t)$ and $Z_i^m(t)$ maintained by an algorithm satisfy the following constraints for all frames $t \in \{0, 1, 2, \ldots\}$.*

$$Q_i^m(t) \leq (Q_i^m)^{max} \text{ and } Z_i^m(t) \leq (Z_i^m)^{max}, \tag{3.6}$$

where $(Q_i^m)^{max}$ and $(Z_i^m)^{max}$ are finite constants. Then the maximum delay of packets can be bounded with a finite constant $(W_i^m)^{max}$, i.e., a packet will be either transmitted or dropped within $(W_i^m)^{max}$. If packets are served in the first-in-first-out (FIFO) manner, according to the ϵ–persistence queue analysis in [86], the delay bound can be written as

$$(W_i^m)^{max} = \lceil ((Q_i^m)^{max} + (Z_i^m)^{max})/\epsilon_i^m \rceil, \tag{3.7}$$

where $\lceil \cdot \rceil$ is the ceiling function.

3.4.2 Lyapunov Optimization

Let $\Theta^m(t)$ be a vector of all $Q_i^m(t)$ and $Z_i^m(t)$, $i \in \mathcal{U}^m$. We define the *Lyapunov function* $L(\Theta^m(t))$ as

$$L(\Theta^m(t)) \doteq \frac{1}{2} \sum_{i \in \mathcal{U}^m} \{(Q_i^m(t))^2 + (Z_i^m(t))^2\}. \tag{3.8}$$

We also define a 1-step sample path *Lyapunov drift* as

$$\Delta_1(\Theta^m(t)) \doteq L(\Theta^m(t+1)) - L(\Theta^m(t)). \tag{3.9}$$

The *drift-plus-penalty* used in Lyapunov optimization [86] is obtained by adding the penalty of spectrum bidding cost. The penalty includes the payments and cost of dropped packets as

$$-V^m \phi^m(t) \doteq V^m b_c^m(t) + V^m \sum_{i \in \mathcal{U}^m} \beta_i^m d_i^m(t), \tag{3.10}$$

where $V^m > 0$ indicates BS m's concern on the price it needs to pay, and β_i^m is the penalty of dropping a packet of UE i, $i \in \mathcal{U}^m$. Hence, the 1-frame *drift-plus-penalty* can be written as $\Delta_1(\Theta^m(t)) + V^m b_c^m(t) + \sum_{i \in \mathcal{U}^m} V^m \beta_i^m d_i^m(t)$. If BS m bids for transmission opportunity on channel c at frame t, the problem can be formulated as follows.

$$\min: \ \Delta_1(\Theta^m(t)) + V^m b_c^m(t) + \sum_{i \in \mathcal{U}^m} V^m \beta_i^m d_i^m(t) \tag{3.11}$$

$$\text{s.t.} \sum_{i \in \mathcal{U}^m} \varphi_{ic}^m(t) = \varphi, \ \text{for } c \in \mathcal{C} \tag{3.12}$$

$$\varphi_{ic}^m(t) \geq 0, \ \text{for } i \in \mathcal{U}^m, c \in \mathcal{C} \tag{3.13}$$

$$R_{ic}^m(t) + R_i^m(t) + d_i^m(t) \leq Q_i^m(t), \ \text{for } i \in \mathcal{U}^m, c \in \mathcal{C} \tag{3.14}$$

$$\epsilon_i^m \geq (A_i^m)^{max}, \ \text{for } i \in \mathcal{U}^m \tag{3.15}$$

$$(d_i^m)^{max} \geq (A_i^m)^{max}, \ d_i^m(t) \geq 0, \ \text{for } i \in \mathcal{U}^m, \tag{3.16}$$

where φ is the total amount of RBs on channel c. In the formulation, (3.12) and (3.13) are resource allocation constraints, while constraint (3.14) guarantees that the packets transmitted and dropped in slot t is no greater than $Q_i^m(t)$.

We can reformulate the *drift-plus-penalty* as follows.

$$\Delta_1(\Theta(t)) + V^m b_c^m(t) + \sum_{i \in \mathcal{U}^m} V^m \beta_i^m d_i^m(t) \tag{3.17}$$

$$\leq B^m + V^m b_c^m(t) + \sum_{i \in \mathcal{U}^m} V^m \beta_i^m d_i^m(t) -$$

$$\sum_{i \in \mathcal{U}^m} Q_i^m(t)(R_{ic}^m(t) + R_i^m(t) + d_i^m(t) - A_i^m(t)) +$$

$$
\begin{aligned}
&\sum_{i\in\mathcal{U}^m} Z_i^m(t)\epsilon_i^m 1_{\{Q_i^m(t)>0\}} - \frac{1}{2}(Z_i^m(t))^2 1_{\{Q_i^m(t)=0\}} - \\
&\sum_{i\in\mathcal{U}^m} Z_i^m(t)(R_{ic}^m(t) + R_i^m(t) + d_i^m(t)) \\
&= B^m - \frac{1}{2}(Z_i^m(t))^2 1_{\{Q_i^m(t)=0\}} + \sum_{i\in\mathcal{U}^m} Q_i^m(t)A_i^m(t)) + \\
&\sum_{i\in\mathcal{U}^m} Z_i^m(t)\epsilon_i^m 1_{\{Q_i^m(t)>0\}} - \Phi_{(1)}^m(t) - \Phi_{(2)}^m(t),
\end{aligned}
$$

where

$$
\begin{cases}
\Phi_{(1)}^m(t) = \sum_{i\in\mathcal{U}^m}(R_{ic}^m(t) + R_i^m(t))(Q_i^m(t) + \\
\qquad Z_i^m(t)) - V^m b_c^m(t) \\
\Phi_{(2)}^m(t) = \sum_{i\in\mathcal{U}^m} d_i^m(t)(Q_i^m(t) + Z^m(t) - V^m\beta_i^m) \\
B^m \doteq \frac{1}{2}\sum_{i\in\mathcal{U}^m}\{[(R_{ic}^m + R_i^m + d_i^m)^{max}]^2 + \\
\qquad 2[(A_i^m)^{max}]^2 + [(\epsilon^m - R_{ic}^m - R_i^m - d_i^m)^{max}]^2\}.
\end{cases}
\tag{3.18}
$$

With Lyapunov optimization [86], we can derive an online algorithm to minimize the *drift-plus-penalty*, which will yield policies for resource allocation, valuation of spectrum, and packet dropping.

Resource Allocation: Maximizing $\Phi_{(1)}^m(t)$ defined in (3.18), we can derive the optimal allocation of RBs and obtain the transmission policy. Note that the first term in $\Phi_{(1)}^m(t)$ is valid only when BS m wins the auction and makes the payment. And the value of the second term does not affect the maximization of $\Phi_{(1)}^m(t)$. We thus solve the following problem.

$$
\begin{aligned}
\max: \quad & \sum_{i\in\mathcal{U}^m}(R_{ic}^m(t) + R_i^m(t))(Q_i^m(t) + Z_i^m(t)) \qquad (3.19)\\
\text{s.t.} \quad & \text{Constraints } (3.12),\ (3.13),\ (3.14).
\end{aligned}
$$

The objective function (3.19) can be rewritten as

$$\begin{aligned} & \sum_{i \in \mathcal{U}^m} (R_{ic}^m(t) + R_i^m(t))(Q_i^m(t) + Z_i^m(t)) \\ = & \sum_{i \in \mathcal{U}^m} \varphi_{ic}^m(t) e_{ic}^m(t)(Q_i^m(t) + Z_i^m(t)) + \\ & \sum_{i \in \mathcal{U}^m} R_i^m(t)(Q_i^m(t) + Z_i^m(t)). \end{aligned} \tag{3.20}$$

Recall that $\varphi_{ic}^m(t)$ is the number of RBs in spectrum c allocated to UE i by BS m. We focus on resource allocation on the unlicensed spectrum and do not consider optimization of the rate from licensed band (i.e., $R_i^m(t)$). Hence we can tune $\varphi_{ic}^m(t)$ to maximize (3.19). Specifically, we apply a greedy algorithm to allocate more RBs to UE i with a higher $e_{ic}^m(t)(Q_i^m(t) + Z_i^m(t))$ under constraints (3.12)–(3.14).

True Value of Channel: To find the highest price that BS m is willing to pay for unlicensed channel c, i.e., $\tilde{b}_c^m(t)$, we can compare $\Phi_{(1)}^m(t)$ when a bid is successful for spectrum c, with that when no bid is made. Since $\tilde{b}_c^m(t)$ is the highest price that BS m is willing to pay for channel c, it is also the *true value* of channel c to BS m.

If the bid is successful, we have

$$\begin{aligned} & \Phi_{(1)}^m(t)' \\ = & \sum_{i \in \mathcal{U}^m} (R_{ic}^m(t) + R_i^m(t))(Q_i^m(t) + Z_i^m(t)) - V^m b_c^m(t). \end{aligned} \tag{3.21}$$

Otherwise, if BS m does not bid for channel c, we have

$$\Phi_{(1)}^m(t)'' = \sum_{i \in \mathcal{U}^m} R_i^m(t)(Q_i^m(t) + Z_i^m(t)). \tag{3.22}$$

When BS m pays the highest price, we have $\Phi^m_{(1)}(t)' - \Phi^m_{(2)}(t)'' = 0$, from which we can solve for $\tilde{b}^m_c(t)$ as

$$\begin{aligned}\tilde{b}^m_c(t) = \frac{1}{V^m} \Bigg\{ & \max \sum_{i \in \mathcal{U}^m} (R^m_{ic}(t) + R^m_i(t)) \times \\ & (Q^m_i(t) + Z^m_i(t)) - \sum_{i \in \mathcal{U}^m} R^m_i(t)(Q^m_i(t) + Z^m_i(t)) \Bigg\} \\ = \frac{1}{V^m} & \max \left\{ \sum_{i \in \mathcal{U}^m} R^m_{ic}(t)(Q^m_i(t) + Z^m_i(t)) \right\} \\ \text{s.t. } & \text{Constraints } (3.12),\ (3.13),\ (3.14).\end{aligned} \tag{3.23}$$

Packets to Drop: By maximizing $\Phi^m_{(2)}(t)$ defined in (3.18), we can obtain the amount of packets to drop as follows.

$$d^m_i(t) = \begin{cases} (d^m_i)^{max}, & Q^m_i(t) + Z^m_i(t) > V^m \beta^m_i \\ 0, & \text{Otherwise}, \end{cases} \tag{3.24}$$

where $(d^m_i)^{max}$ is a constant, i.e., a predefined limit for d^m_i. To satisfy the maximum delay requirement, packets are dropped as in (3.24) in each frame, whether or not there is addition transmission opportunity on unlicensed bands.

3.4.3 Guarantee On Maximum Delay

In this section, we first derive upper bounds on the real and virtual queue lengths. We then translate the backlog bounds to an upper bound on queueing delay.

Lemma 3.1. *With the drop decision (3.24) and assuming* $0 \leq \epsilon^m_i \leq (d^m_i)^{max}$ *and* $0 \leq (A^m_i)^{max} \leq (d^m_i)^{max}$*, the proposed resource allocation and dropping policies ensure the following upper bounds on the real and virtual queues.*

$$(Q^m_i(t) + Z^m_i(t))^{max} = V^m \beta^m_i + (A^m_i)^{max} + \epsilon^m_i \tag{3.25}$$

$$(Z_i^m)^{max} = V^m \beta_i^m + \epsilon_i^m. \tag{3.26}$$

Proof. We first prove (3.25) with *induction*. Since the real and virtual queues are all initially empty, we have $Q_i^m(0) + Z_i^m(0) \leq V^m \beta_i^m + (A_i^m)^{max} + \epsilon_i^m$. Then we assume (3.25) holds for some $t_0 \geq 0$, and prove that (3.25) also holds for $(t_0 + 1)$.

If $Q_i^m(t_0) + Z_i^m(t_0) \leq V^m \beta_i^m$, it follows (3.2) and (3.5) that

$$\begin{aligned} & Q_i^m(t_0+1) + Z_i^m(t_0+1) \\ \leq\ & Q_i^m(t_0) + Z_i^m(t_0) + (A_i^m)^{max} + \epsilon_i^m \\ \leq\ & V^m \beta_i^m + (A_i^m)^{max} + \epsilon_i^m. \end{aligned}$$

Otherwise, if $V^m \beta_i^m \leq Q_i^m(t_0) + Z_i^m(t_0) \leq V^m \beta_i^m + (A_i^m)^{max} + \epsilon_i^m$, then we have $d_i^m(t) = (d_i^m(t))^{max}$ according to (3.24). Hence

$$\begin{aligned} & Q_i^m(t_0+1) + Z_i^m(t_0+1) \\ \leq\ & Q_i^m(t_0) - R_{ic}^m(t_0) - R_i^m(t_0) - (d_i^m)^{max} + A_i^m(t_0) + \\ & Z_i^m(t_0) + \epsilon_i^m - R_{ic}^m(t_0) - R_i^m(t_0) - (d_i^m)^{max} \\ \leq\ & Q_i^m(t_0) + Z_i^m(t_0) + A_i^m(t_0) + \epsilon_i^m - 2(d_i^m)^{max} \\ \leq\ & V^m \beta_i^m + (A_i^m)^{max} + \epsilon_i^m. \end{aligned}$$

Thus (3.25) also holds for the case of $(t_0 + 1)$, and we conclude that (3.25) is true for all t. The proof for (3.26) is similar to that in [86] and is omitted for brevity. □

Theorem 3.1. *With the proposed resource allocation and packet dropping polices and the FIFO service discipline, the queueing delay is upper bounded by* $(W_i^m)^{max}$*. That is, any packet is either transmitted or dropped within* $(W_i^m)^{max}$*, given by*

$$(W_i^m)^{max} = 2 + (2V^m \beta_i^m + (A_i^m)^{max})/\epsilon_i^m. \tag{3.27}$$

Proof. According to Fact 1, we have

$$(W_i^m)^{max} = ((Q_i^m)^{max} + (Z_i^m)^{max})/\epsilon_i^m.$$

It follows Fact 3.1 and Lemma 3.1 that

$$\begin{aligned}(W_i^m)^{max} &\leq ((Q_i^m + Z_i^m))^{max} + (Z_i^m)^{max})/\epsilon_i^m \\ &= 2 + (2V^m\beta_i^m + (A_i^m)^{max})/\epsilon_i^m.\end{aligned}$$

□

From Theorem 3.1, we see that there is a approximately linear relationship between the maximum delay and $V^m\beta_i^m/\epsilon_i^m$.

3.5 Auction and Pricing

3.5.1 Determine The Auction Winner

During the auction, the same spectrum can only be allocated to a set of BS's with no mutual interference at a time. A set of BS's with no mutual interference can be denoted as a non-interfering bidding set. In each auction, the auction session determines the bidding set $\alpha_c^{m^*}(t)$ that wins the auction and obtains the opportunity of transmission in frame t. The objective of the auction is to maximize the sum of bids in $\alpha_c^{m^*}(t)$. It follows that

$$\max_{\{\alpha_c^{m^*}(t)\}} : \ G_c(t)|_{\{S_c^{m^*}(t)\}} \doteq \sum_{m \in \alpha_c^{m^*}(t)} b_c^m(t) \tag{3.28}$$

$$\text{s.t. } I_{i,j} = 0, \text{for all } i, j \in \alpha_c^{m^*}(t). \tag{3.29}$$

Recall that $S_c^{m^*}(t)$ is the set of BS's that bid in the auction. Constraint (3.29) guarantees that there is no mutual interference among the winner BS's. It is possible that the solution to problem (3.28)

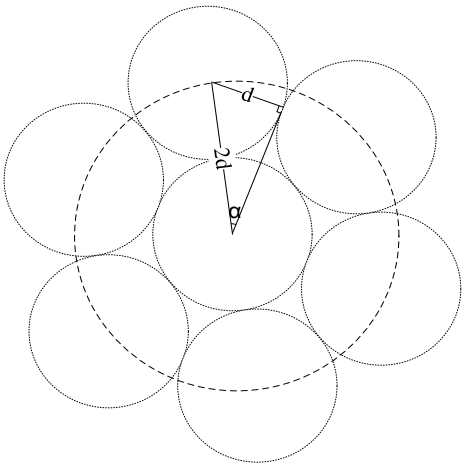

Figure 3.2: Illustrate the maximum independent set.

is not unique. Then the auction session will randomly choose one non-interfering set as a winner set.

To solve problem (3.28), we could use a brute-force approach to examine all the BS combinations for the optimal non-interfering bidding set, which has a complexity of $\mathcal{O}(2^n - 1)$, where $n = |S_c^{m^*}(t)|$ is the number of BS's participating in the auction. In fact, this is a *maximum weighted independent set problem* in graph theory [81], which is NP-complete. Wu *et al.* in [91] proposed an approximation solution with a polynomial complexity by relaxing the objective function. Fortunately, in the auction design of this work, the number of BS's in a non-interfering bidding set is limited. Without loss of generality, if the interference range of each BS is shaped as disks with an identical diameter, the maximum number of BS's in a non-interfering bidding set is 7. The proof is given below.

Recall that all the bidding BS's are in the interference range of the auction initiator. Hence the distance between any two BS's is no more than $2d$, where d is the interference range of a BS. Without loss of generality, we assume that the interference range of each BS is shaped as a disk with diameter d. Then a maximal independent bidding set should be formed as shown in Fig. 3.2. As $\alpha = \pi/6$, there are 6 disks in the outer layer and the size of the independent set is 7.

We propose a recursive algorithm WINNERSET$\big(S_c^{m^*}(t)\big)$ to solve problem (3.28), to find the maximum sum of bids of a non-interfering bidding set. The algorithm has a complexity of $\mathcal{O}(n(k-$

Algorithm 2: WINNERSET$\left(S_c^{m^*}(t)\right)$

Input: Information about participating BS's, $S_c^{m^*}(t)$ and $I_{i,j}(t)$, and all the bids made for channel c, $b_c^m(t)$, for $i, j, m \in S_c^{m^*}(t)$

Output: The Optimum Non-interfering Bidding Set

```
G^max = 0 ;
α = ∅ ;
for each BS m ∈ S_c^{m*}(t) do
    S = S_c^{m*}(t) ;
    Delete BS m and all BS's interfering with BS m from S ;
    α' = WINNERSET(S) ∪ m ;
    Compute G' as the sum of all bids in α' ;
    if G' > G^max then
        G^max = G' ;
        α = α' ;
    end
end
Return α ;
```

$1)!)$, where k is the maximum depth of the recursive algorithm, which is equal to the maximum number of BS's in a non-interfering bidding set.

As shown in Algorithm 2, the recursive algorithm WINNERSET$(\cdot)$ works as follows. The goal is to obtain the maximum non-interfering bidding set among all the sets that contain BS m, for $m \in S_c^{m^*}(t)$. The maximum non-interfering bidding set containing BS m, is BS m plus the maximum non-interfering bidding set α' in $S_c^{m^*}(t)$, after deleting BS m and all its interfering BS's. And α' can be obtained recursively.

In our auction design, all bidders are equal. Hence, we introduce the second-price strategy in second-price sealed-bid auctions (i.e., *Vickrey auctions*) [89, 91], in which the auction winner pays the second highest bid among the bidders. Applying this strategy, the winning BS set $\alpha_c^{m*}(t)$ pays for the maximum sum bids of the non-interfering bidding sets among the losers(the maximum independent set aside from the winner set, denoted as secondary winner set). Different from the traditional second-price strategy, there may be multiple winners in a single auction in our design.

Hence we need to split the payment among the winners, given by

$$\sum_{m\in\alpha_c^{m^*}(t)} \hat{b}_c^m(t) = G_c(t)|_{\{S_c^{m^*}(t)\backslash\alpha_c^{m^*}(t)\}}, \tag{3.30}$$

where $G_c(t)|_{\{S_c^{m^*}(t)\backslash\alpha_c^{m^*}(t)\}}$ is the maximum sum bids of the non-interfering bidding sets among the losers.

To effectively split the payment among winners, a Nash bargaining solution (NBS) is introduced in [91], aiming to maximize $\sum_{m\in\alpha_c^{m^*}(t)}(b_c^m(t) - \hat{b}_c^m(t))$. However, the solution in [91] ignores the constraint $b_c^m(t) - \hat{b}_c^m(t) \geq 0$, for $m \in \alpha_c^{m^*}(t)$. Actually, we could obtain a truthful bidding if $0 \leq \hat{b}_c^m(t) \leq b_c^m(t)$ (as given by Theorem 3.2 in Section 3.5.2). Hence we propose the following pricing scheme.

$$\hat{b}_c^m(t) = \begin{cases} b_c^m(t)\frac{G_c(t)|_{\{S_c^{m^*}(t)\backslash\alpha_c^{m^*}(t)\}}}{G_c(t)|_{\{S_c^{m^*}(t)\}}}, & m \in \alpha_c^{m^*}(t) \\ -b_c^m(t), & m \in \alpha_c^{m^*}(t)' \\ 0, & \text{otherwise}, \end{cases} \tag{3.31}$$

where $\alpha_c^{m^*}(t)'$ is the optimal set of non-interfering loser BS's in $S_c^{m^*}(t)\backslash\alpha_c^{m^*}(t)$.

3.5.2 Proposed LMWA Algorithm And Performance Analysis

With the proposed schemes for resource allocation, valuation of spectrum, packet dropping, and auction, we develop an integrated algorithm for the LTE-unlicensed system, named Lyapunov based Multi-Winner Auction (LMWA), which is presented in Algorithm 3. In Line 10 of LMWA, no bid would be made to avoid the hidden node problem.

We have the following theorems on the performance of LMWA about truthful bidding, utility and social welfare maximization, and the QoS of UEs.

Theorem 3.2. *(Truthful Bidding) The pricing scheme in (3.31) guarantees the truthfulness of bidding, i.e.,* $b_c^m(t) = \tilde{b}_c^m(t)$.

Algorithm 3: The Proposed LMWA Algorithm

for *each BS* m *idle on unlicensed bands* **do**
  if *a set of channels* $\mathcal{C}'$ *on unlicensed bands are sensed idle at frame* t **then**
    Randomly select a channel c from $\mathcal{C}'$;
    Compute $R_{ic}^m(t)$ as in (3.19), $\tilde{b}_c^m(t)$ as in (3.23), and $d_i^m(t)$ as in (3.24) ;
    if *a BS in the interference range is the auction initiator of channel* c **then**
      Submit $b_c^m(t) = \tilde{b}_c^m(t)$ to the auction initiator ;
    else if *no BS in the interference range is bidding for* c **then**
      BS m becomes the auction initiator and broadcasts a message to hold channel c ;
    else
      Continue ;
    end
  end
end
Each auction session decides the winner BS set with Alg. 2 ;
Each auction session decides the actual price $\hat{b}_c^m(t)$ as in (3.31) ;
for *each BS* m*, at the beginning of frame* t **do**
  Drop $d_i^m(t)$ packets as in (3.24) in frame t ;
  if *BS* m *wins a bid* **then**
    Schedule transmission on channel c with $R_{ic}^m(t)$ in frame t ;
  end
end

Proof. With the proposed pricing scheme (3.31), the payment of a winner $\hat{b}_c^m(t)$ is a complicated function of bid $b_c^m(t)$. It also depends on other BS's bids, which are unknown to BS m before submitting its bid. Hence a bidder cannot predict the payment during the auction. If $b_c^m(t) > \tilde{b}_c^m(t)$, then it may be charged with a price $\hat{b}_c^m(t) > \tilde{b}_c^m(t)$. If $b_c^m(t) < \tilde{b}_c^m(t)$, then it has a lower chance to win the auction. Hence, $b_c^m(t) = \tilde{b}_c^m(t)$ is always the best bidding strategy.

The proposed pricing scheme (3.31) also resistant to the version of shill bidding in which a buyer uses multiple identities in the auction in order to maximize its profit [108]. In shill bidding, one identity of a buyer submit a price high enough to surely win the auction and the another identity of the same buyer submit a price high enough to be the second highest price. In this case, the buyer will win the auction and only pay to itself. In this work, two or more BSs from the same operators may form multiple identities of the buyer(the operator). However, in pricing scheme (3.31), BSs from the same operator have no clue of whether there would be any other BS/BSs in the secondary winner set without overall interfering matrix and bids from other BSs. If they apply the shill

bidding in [108] and there is any other BS in the secondary winner set, they would need to make a high payment to other BS/BSs in the secondary winner set. □

Theorem 3.3. *(Utility Maximization for Individual BS) If the compound process* $\{A_i^m(t), e_{ic}^m(t)\}$ *is i.i.d. over frames and for any UE* i *served by BS* m*, the proposed LMWA algorithm achieves the following lower bound on the utility of BS* m*.*

$$\mathbb{E}\{\phi_c^m(t)\} \geq \{\phi_c^m\}^{opt} - B^m/V^m, \tag{3.32}$$

where $\phi_c^m(t)$ *is the utility of BS* m *defined in (3.3),* B^m *is defined in (3.18), and* $(\phi_c^m)^{opt}$ *is the maximum utility BS* m *can achieve without knowing the bids of others in an auction.*

Proof. According to (3.17), we have

$$\begin{aligned} &\Delta_1(\Theta(t)) + V^m b_c^m(t) + \sum_{i\in\mathcal{U}^m} V^m \beta_i^m d_i^m(t) \qquad (3.33)\\ \leq\ & B^m + V^m b_c^m(t) + \sum_{i\in\mathcal{U}^m} V^m \beta_i^m d_i^m(t) + \\ & \sum_{i\in\mathcal{U}^m} Q_i^m(t)(A_i^m(t) - R_{ic}^m(t) - R_i^m(t) - d_i^m(t)) + \\ & \sum_{i\in\mathcal{U}^m} Z_i^m(t)(\epsilon_i^m 1_{\{Q_i^m(t)>0\}} - R_{ic}^m(t) - R_i^m(t) - d_i^m(t)). \end{aligned}$$

Then for any (possibly randomized) feasible schedule, we have

$$\begin{aligned} &\min\{\Delta_1(\Theta(t)) - V^m \phi_c^m(t)\} \qquad (3.34)\\ \leq\ & B^m + V^m \phi_c^{m*}(t) + \\ & \sum_{i\in\mathcal{U}^m} Q_i^m(t)(A_i^m(t) - R_{ic}^{m*}(t) - R_i^{m*}(t) - d_i^{m*}(t)) + \\ & \sum_{i\in\mathcal{U}^m} Z_i^m(t)(\epsilon_i^m 1_{\{Q_i^m(t)>0\}} - R_{ic}^{m*}(t) - R_i^{m*}(t) - d_i^{m*}(t)), \end{aligned}$$

where $R_{ic}^{m*}(t)$, $R_i^{m*}(t)$, and $d_i^{m*}(t)$ are the terms corresponding to the feasible schedule. Now we consider a randomized scheduling policy that achieves the following for each application $i \in \mathcal{U}^m$.

$$\mathbb{E}\{\phi_c^{m*}(t)\} = \{\phi_c^m(t)\}^{opt} \tag{3.35}$$

$$\mathbb{E}\{A_i^m(t) - R_{ic}^{m*}(t) - R_i^{m*}(t) - d_i^{m*}(t)\} \leq 0 \tag{3.36}$$

$$\mathbb{E}\{\epsilon_i^m 1_{\{Q_i^m(t)>0\}} - R_{ic}^{m*}(t) - R_i^{m*}(t) - d_i^{m*}(t)\} \leq 0, \tag{3.37}$$

where $\{\phi_c^m(t)\}^{opt}$ is the maximum utility BS m can achieve in a stable system, and (3.36) and (3.37) stabilize the queues.

Hence, as the proposed LMWA algorithm minimizes (3.34), we have

$$\begin{aligned}
&\mathbb{E}\{\Delta_1(\Theta(t)) - V^m \phi_c^m(t)|t\} \leq B^m - V^m\{\phi_c^m(t)\}^{opt} + \\
&\mathbb{E}\left\{\sum_{i\in\mathcal{U}^m} Q_i^m(t)(A_i^m(t) - R_{ic}^{m*}(t) - R_i^{m*}(t) - d_i^{m*}(t))\right\} + \\
&\mathbb{E}\left\{\sum_{i\in\mathcal{U}^m} Z_i^m(t)(\epsilon_i^m 1_{\{Q_i^m(t)>0\}} - R_{ic}^{m*}(t) - R_i^{m*}(t) - \right. \\
&\left. d_i^{m*}(t))\right\} \leq B^m - V^m\{\phi_c^m(t)\}^{opt},
\end{aligned}$$

where

$$\begin{aligned}
&\mathbb{E}\left\{\sum_{i\in\mathcal{U}^m} Q_i^m(t)(A_i^m(t) - R_{ic}^{m*}(t) - R_i^{m*}(t) - d_i^{m*}(t))\right\} \leq 0 \\
&\mathbb{E}\left\{\sum_{i\in\mathcal{U}^m} Z_i^m(t)(\epsilon_i^m 1_{\{Q_i^m(t)>0\}} - R_{ic}^{m*}(t) - R_i^{m*}(t) - \right. \\
&\left. d_i^{m*}(t))\right\} \leq 0.
\end{aligned}$$

Then we have

$$\mathbb{E}\{\Delta_1(\Theta(t)) - V^m \phi_c^m(t)|t\} \leq B^m - V^m\{\phi_c^m(t)\}^{opt}$$

for the proposed LMWA algorithm. Notice that $\sum_{t=0}^{T-1}\mathbb{E}\{\Delta_1(\Theta(t))|t\} = \mathbb{E}\{L(\Theta(t))\} < \infty$ for a stable system. It follows that

$$\begin{aligned}
&\limsup_{t\to\infty}\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\{\Delta_1(\Theta(t))|t\} - \limsup_{t\to\infty}\frac{V^m}{T}\sum_{k=0}^{T-1}\mathbb{E}\{\phi_c^m(t)\}\\
&= 0 - \limsup_{t\to\infty}\frac{V^m}{T}\sum_{k=0}^{T-1}\mathbb{E}\{\phi_c^m(t)\}\\
&\leq B^m - V^m\{\phi_c^m(t)\}^{opt}.
\end{aligned}$$

Then we conclude that Theorem 3.3 holds true. □

It follows that with LMWA, each BS can achieve an average utility with a gap of B^m/V from the optimal average utility.

Theorem 3.4. *(Social Welfare Maximization or Weighted Dropping Minimization) If $V^m \doteq V$ is a constant for all BS's, and the compound process $\{A_i^m(t), e_i^m(t)\}$ is i.i.d. over frames, for BS $m \in S_c^{m^*}(t)$ and UE $i \in \mathcal{U}^m$, then for each auction the following inequality holds true.*

$$\begin{aligned}
&\sum_{m\in S_c^{m^*}(t)}\sum_{i\in\mathcal{U}^m}\mathbb{E}\{\beta_i^m d_i^m(t)\} \qquad (3.38)\\
&\leq \mathbb{E}\left\{\sum_{m\in S_c^{m^*}(t)}\sum_{i\in\mathcal{U}^m}[\beta_i^m d_i^m(t)]\right\}^{opt} + B/V,
\end{aligned}$$

where $B = \sum_{m\in S_c^{m^*}(t)} B^m$, B^m is given in (3.18), and $\mathbb{E}\{\sum_{m\in S_c^{m^*}(t)}\sum_{i\in\mathcal{U}^m}[\beta_i^m d_i^m(t)]\}^{opt}$ is the expected minimum weighted dropping penalty that can be achieved in an auction.

Proof. As in (3.28), the proposed LMWA algorithm maximizes $\sum_{m\in\alpha_c^{m^*}(t)} b_c^m(t)$ in the auction part. According to (3.23) and Theorem 3.2, we have

$$\sum_{m\in\alpha_c^{m^*}(t)} b_c^m(t) = \sum_{m\in\alpha_c^{m^*}(t)} \tilde{b}_c^m(t)$$

$$= \frac{1}{V^m} \sum_{m \in \alpha_c^{m^*}(t)} \max \left\{ \sum_{i \in \mathcal{U}^m} R_{ic}^m(t)(Q_i^m(t) + Z_i^m(t)) \right\}.$$

As R_{ic}^m, $Q_i^m(t)$, and $Z_i^m(t)$ are independent among different BS's at frame t, LMWA maximizes $\sum_{m \in \alpha_c^{m^*}(t)} \sum_{i \in \mathcal{U}^m} \{R_{ic}^m(t) + R_i^m(t)\}(Q_i^m(t) + Z_i^m(t))$ in each auction, by enforcing the constraint that no interfering BS's transmit at the same time. Based on theorem(3.3) we have

$$\begin{aligned} & \mathbb{E} \left\{ \sum_{m \in S_c^{m^*}(t)} \sum_{i \in \mathcal{U}^m} \left[-\hat{b}_i^m(t) - \beta_i^m d_i^m(t) \right] \right\} \\ & \geq \mathbb{E} \left\{ \sum_{m \in S_c^{m^*}(t)} \sum_{i \in \mathcal{U}^m} \left[-\hat{b}_i^m(t) - \beta_i^m d_i^m(t) \right] \right\}^{opt} - \frac{B}{V}. \end{aligned} \tag{3.39}$$

Since $\sum_{m \in S_c^{m^*}(t)} \hat{b}_i^m(t) = 0$ according to the auction design, the above inequality (3.39) can be simplified as

$$\begin{aligned} & - \sum_{m \in S_c^{m^*}(t)} \sum_{i \in \mathcal{U}^m} \mathbb{E}\{\beta_i^m d_i^m(t)\} \\ & \geq \mathbb{E} \left\{ \sum_{m \in S_c^{m^*}(t)} \sum_{i \in \mathcal{U}^m} [-\beta_i^m d_i^m(t)] \right\}^{opt} - \frac{B}{V}. \end{aligned} \tag{3.40}$$

Thus we conclude that (3.38) holds true. □

In the special case with $\beta_i^m = \beta$ for all UEs and BS's involved in the auction, we have

$$\begin{aligned} & \sum_{m \in S_c^{m^*}(t)} \sum_{i \in \mathcal{U}^m} \mathbb{E}\{d_i^m(t)\} - \mathbb{E} \left\{ \sum_{m \in S_c^{m^*}(t)} \sum_{i \in \mathcal{U}^m} [d_i^m(t)] \right\}^{opt} \\ & \leq B/(V\beta). \end{aligned} \tag{3.41}$$

In this special case, it can be seen that the optimality gap for packet drop rate is proportional to $1/(V\beta)$. If $V\beta \to \infty$, the proposed LMWA algorithm can achieve the minimum drop rate in each

auction. Furthermore, according to Theorem 3.1, the maximum delay is proportional to $V\beta$. There is clearly a tradoff between packet drop and delay here.

3.6 Simulation Validation

In this section, we use Matlab simulations to evaluate the performance of the proposed algorithms with a typical outdoor small cell scenario. We used two simple schemes as benchmarks: (i) Single-Winner that selects only one winner during an auction; and (ii) Random Access that randomly selects a winner during the bidding stage. The configuration of simulation parameters is based on [85], as summarized in Table 3.1. Specifically, we set $\epsilon_i^m = 8$ and $(d_i^m)^{max} = 8$ for all UEs, which are both normalized to the time scale of one second. We also set $\beta_i^m = \beta$ to better reveal its impact. The network area of 200×200 m^2 is covered with LTE macro cells in licensed bands and the average data rate provided by the LTE Macro cell is 4 Mbytes/s for all UES. Six LTE-unlicensed BS's are deployed in the area, each serving 10 UEs. Two channels on the LTE-unlicensed band are available.

We adopt a truncated Poisson traffic model in the simulations, which is a Poisson process with arrival rate λ and the maximum number of arrival packets is bounded with 2λ. The packet size is 2 Mbytes (a file in the application and can be separated two smaller packets to fit the MAC layer packet size [95]). In this work, we focus on the coordination among LTE-unlicensed users, so the evaluation of WiFi performance is not included.

We also adopted *COST 231 Hata* for metropolitan areas as the propagation model [106]. where

$$\begin{aligned} L(d) = 46.3 + 33.9\log_{10}(f_c) - 13.82\log_{10}(h_b) - \alpha(h_m) \\ + (44.9 - 6.55\log_{10}(h_b))\log_{10}(d) + 3 \end{aligned} \tag{3.42}$$

where $\alpha(h_m) = 3.2(\log_{10}(11.75))^2 - 4.75$, f_c, h_b and h_m are the central frequency, height of BS and height of mobile device respectively.

Table 3.1: Simulation Parameters

Parameter	*Scenario for Outdoor Small Cells*
Carrier frequency	5 GHz
Bandwidth	20 MHz
Number of RBs	100
Frame duration	10 ms
BS height	10 m, below rooftop
mobile device height	1.5m
Antenna configuration	2Tx-2Rx
Transmit power	30 dBm
UE noise figure	7 dB
Channel model	UMi outdoor
BS antenna configuration	Omni-directional, 0 dBi gain
UE antenna gain	0 dBi
Thermal noise	-174 dBm/Hz
LBT threshold	-85 dBm
Traffic model	Poisson
Packet size	2 Mbytes

In Fig. 3.3, we present the relationship between arrival rate of packets and the average packet dropping rate. We find that the average dropping rate is increasing as the arrival rate grows. The proposed LMWA algorithm outperforms the two other schemes with a considerably smaller dropping rate. This is because that under the proposed LMWA algorithm, spectrum in unlicensed bands can be spatially reused, and the lower dropping rate is enabled by an higher throughput due to spectrum reuse. We can also see that the dropping rate of Single-Winner is also considerably lower that of Random Access, which indicates that the auction enables the BS with a higher utility to win the unlicensed spectrum.

In Fig. 3.4, we present the relationship between packet arrival rate and average queueing delay. The simulation shows there is a linear relationship between the arrival rate and average delay, thus validating Theorem 3.1. The increased arrival rate do not cause a surge in delay. Hence, even if the arrival rate is really high, LMWA can still guarantee that the QoS requirement that a packet is dropped or transmitted within a limited time. As in the previous case, the proposed LMWA

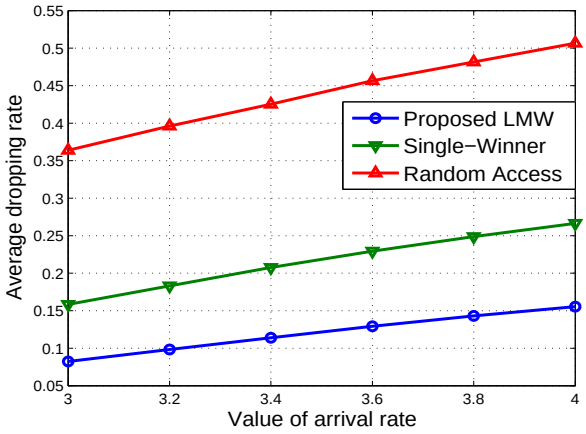

Figure 3.3: Packet arrival rate versus average drop rate: $V\beta = 20$ for all UEs.

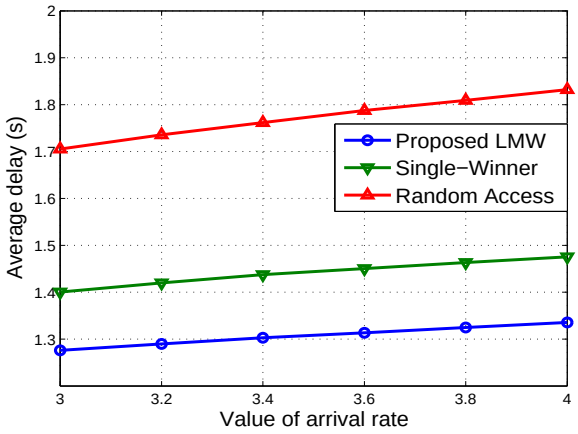

Figure 3.4: Packet arrival rate versus average delay: $V\beta = 20$ for all UEs.

algorithm outperforms the two benchmarks with considerable gains, while Single-Winner also outperforms Random Access.

In Fig. 3.5, we present the relationship between arrival rate and average throughput. The simulation shows that throughput increases with the increasing of the arrival rate for all three algorithms, while the curve for Random Access is pretty flat.

In Fig. 3.6, we present the relationship between $V\beta$ and the average dropping rate. The simulation confirms the $O(1/V\beta)$ bound of dropping packets. For the proposed LMWA and Single-Winner algorithm, the average dropping rate decreases as $V\beta$ grows, and the $O(1/V\beta)$ bound of dropping packets can be observed. Hence, we can choose $V\beta$ to better tradeoff between the QoS

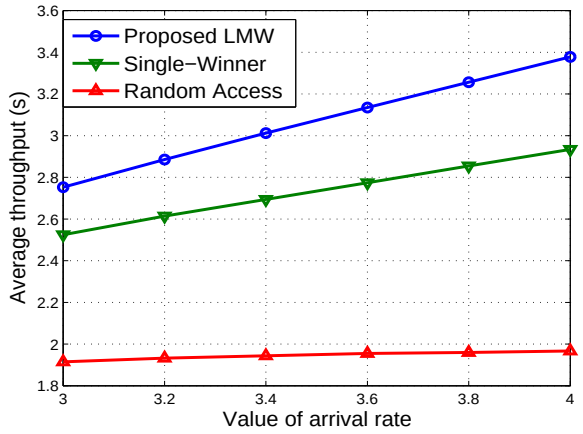

Figure 3.5: Packet arrival rate versus average throughput: $V\beta = 20$ for all UEs.

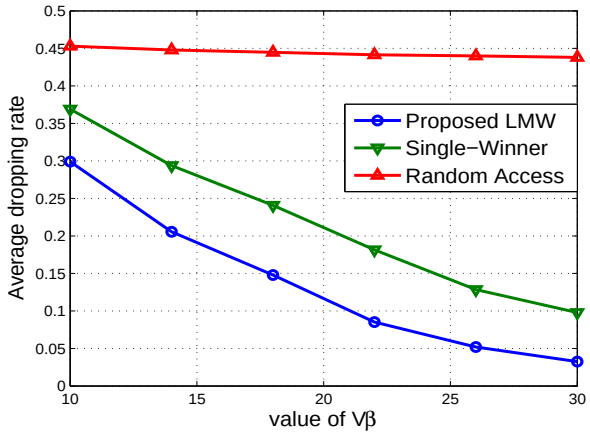

Figure 3.6: $V\beta$ versus average drop rate: $A_i^m = 3.5$ for all UEs.

requirements on dropping rate and delay in practice. For Random Access, we find that the dropping rate does not decrease significantly with increased $V\beta$. This is because that the arrival rate is much higher than the provided throughput and dropping of many packets is unavoidable, even with a loosen delay requirement. Obviously, this simulation shows similar gap among the performance of the three schemes as in Figs. 3.3, 3.4 and 3.5.

In Fig. 3.7, we show the relationship between $V\beta$ and average delay. The simulation confirms the bound of delay and $V\beta$ as in Theorem 3.1. Although Theorem 3.1 is about the upper bound of the maximum delay, we can still see that there is an approximately linear relationship between average delay and $V\beta$ in all the three curves, which all adopt the proposed dropping policy.

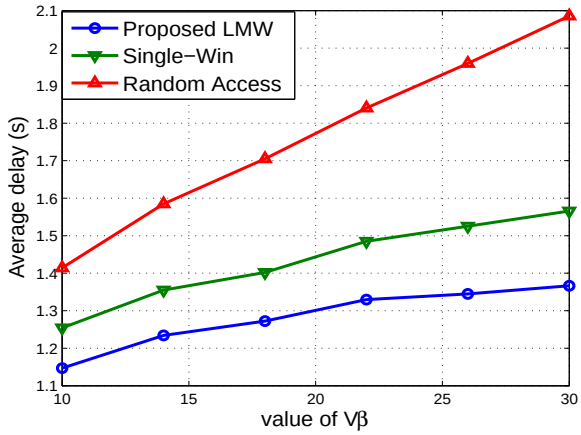

Figure 3.7: $V\beta$ versus average delay: $A_i^m = 3.5$ for all UEs.

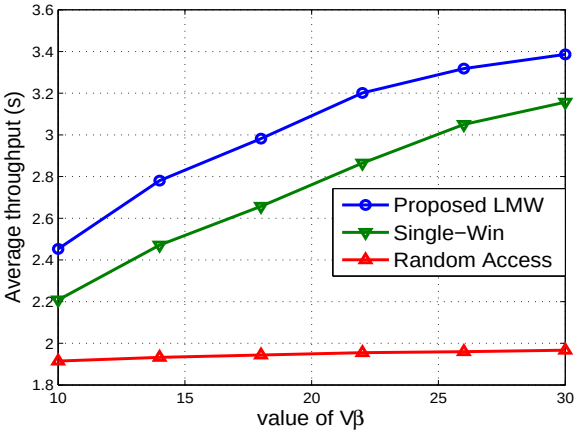

Figure 3.8: $V\beta$ versus average throughput: $A_i^m = 3.5$ for all UEs.

In addition, the proposed algorithm outperforms Single-Winner, and Single-Winner outperforms Random Access in this simulation again.

In Fig. 3.8, we present the relationship between $V\beta$ and average throughput. The simulation shows that throughput increases with the increasing of the arrival rate for all three algorithms and the simulation shows similar gap among the performance of the three schemes Fig.3.5.

3.7 Conclusions

We studied distributed online auction for sharing unlicensed bands among LTE-unlicensed BS's to maximize the social welfare in each auction, while achieving the dual goal of minimizing

the expected packet dropping rate and guarantee a maximum delay. Specifically, we propose Lyapunov optimization based schemes to evaluate the true value of unlicensed spectrum, to allocate RBs on unlicensed bands, and to decide when to drop packets based on current channel condition, queue lengths, and delay of packets. We also proposed a truthful auction mechanism to integrate the schemes, which can maximize the overall social welfare and guarantee bounded drop rate and delay. The superior performance of the proposed algorithms over two benchmark schemes was validated with simulations.

Chapter 4

Online Channel Assignment, Transmission Scheduling, and Transmission Mode Selection in Multi-channel Full-duplex Wireless LANs

4.1 Introduction

Due to the dramatic increase of wireless data demands driven by the wide use of smartphones, tablets and other smart devices, there is an urgent need to improve the spectrum efficiency of existing wireless networks. Through effective self-interference cancellation, full-duplex transmission, i.e., transmitting and receiving simultaneously in the same band, has been successfully demonstrated [36]. With various self-interference cancellation techniques, full-duplex transmission has the potential to increase and even double the wireless link capacity [37].

Combined with RF interference cancellation and digital baseband interference cancellation, antenna cancellation can achieve a sufficient self-interference cancellation for full-duplex transmissions. In [37–39], analog and digital cancellation techniques were investigated. With full-duplex transmissions, various full-duplex links can be formed. For example, in the three-node full-duplex link scenario, one node (e.g., a base station) executes self-interference cancellation to transmit to and receive from two different half-duplex nodes simultaneously [40]. In the two-node link scenario, both nodes are capable of self-interference cancellation and can transmit to and receive from each other simultaneously [41].

Due to imperfect self-interference cancellation, the residual self-interference may still lead to a lower signal-to-interference-plus-noise ratio (SINR) and deteriorate the performance of a full-duplex link [42]. Additional power is needed to combat the residual self-interference to achieve a suitable SINR. As a result, full-duplex transmission may not always be helpful, and there is a trade-off between the energy cost and delay in the design of full-duplex wireless networks [43]. In [42,

43], the extra energy consumption and the limits of full-duplex transmission were investigated. Joint resource allocation and scheduling in wireless networks is a challenging problem, for which Lyapunov optimization has been applied and shown effective [9, 44–46]. However, these prior works are all focused on half-duplex wireless networks. Many challenging issues that arise in full-duplex wireless networks have not been adequately addressed.

In this chapter, we consider a multi-channel wireless LAN (WLAN) where both the access point (AP) and user equipments (UE) are capable of full-duplex transmission. Since full-duplex is not always more efficient than half-duplex, we aim to jointly consider the problems of channel assignment, transmission scheduling, and transmission mode selection for the AP and UEs. We develop a problem formulation to capture the trade-off between energy consumption and queue length (which is indicative of delay) in the multi-channel full-duplex WLAN, with the objective to minimize the overall energy consumption of the system and stabilize the packet queues at all the nodes. We then develop an effective solution algorithm based on the Lyapunov optimization framework. With the proposed algorithm, the overall optimization problem over the entire time period is first reduced to the maximization of a *drift-plus-penalty* for each node in each time slot. The reduced problem only depends on the queue lengths, wireless link rates, and energy consumptions in the current time slot. We then transform the reduced problem into a maximum weighted matching problem and solve it with the Hungarian Method [47].

The proposed algorithm is an online algorithm since it does not require any past and future information of the WLAN system. We prove that the proposed algorithm maximizes the *drift-plus-penalty* among all possible transmission modes and channel assignment schemes. Furthermore, we derive upper bounds on the average sum queue length and average total energy consumption under the proposed algorithm, which clearly demonstrate the energy-delay trade-off in the multi-channel full-duplex WLAN. The performance of the proposed algorithm is validated with simulations.

The remainder of this chapter is organized as follows. The system model and problem formulation are presented in Section 4.2. The proposed scheduling algorithm is developed and analyzed in Section 4.3. A simulation study is presented in Section 4.4. Section 4.5 concludes this work.

4.2 System Model and Problem Statement

4.2.1 System Model

We consider a WLAN with one AP, a set of UEs denoted as $\mathcal{N} = \{1, 2, ..., N\}$, and a set of orthogonal channels denoted as $\mathcal{S} = \{1, 2, ..., S\}$. The AP determines the channel assignment, transmission schedule, and transmission mode selection for both uplink and downlink transmissions. We assume that data is transmitted via the AP in packets, and there is no direct transmission among the UEs. The packets waiting for transmission are buffered and served in the First In First Out (FIFO) manner. We assume a discrete time system. The uplink queue lengths at the beginning of time slot t are denoted as $\vec{Q}^u(t) = \{Q_1^u(t), Q_2^u(t), ..., Q_N^u(t)\}$ and the downlink queue lengths are denoted as $\vec{Q}^d(t) = \{Q_1^d(t), Q_2^d(t), ..., Q_N^d(t)\}$, where $Q_i^u(t)$ is the backlog of the uplink queue maintained at UE i and $Q_i^d(t)$ is the backlog of the downlink virtual queue for UE i maintained at the AP.

At time slot t, the arrivals of packets to the uplink queues are denoted as $\vec{\mathcal{A}}^u(t) = \{A_1^u(t), A_2^u(t), ..., A_N^u(t)\}$. The arrivals of packets to the downlink queues are denoted as $\vec{\mathcal{A}}^d(t) = \{A_1^d(t), A_2^d(t), ..., A_N^d(t)\}$. In addition, we assume that the arrivals of packets, either to the uplink or downlink queues, are i.i.d over time. The expectations, i.e., the average arrival rates, are

$$\vec{\lambda}^u \triangleq \mathbb{E}\{\vec{\mathcal{A}}^u(t)\} = \{\lambda_1^u, \lambda_2^u, ..., \lambda_N^u\} \text{ and } \vec{\lambda}^d \triangleq \mathbb{E}\{\vec{\mathcal{A}}^d(t)\} = \{\lambda_1^d, \lambda_2^d, ..., \lambda_N^d\}. \tag{4.1}$$

Recall that there are $\mathcal{S} = \{1, 2, ..., S\}$ orthogonal channels. During each time slot t, a UE can transmit and/or receive on one of the channels in $\mathcal{S}$. The channel assignment decision is denoted as $\alpha_i(t)$, where $i \in \mathcal{N}$ and $\alpha_i(t) \in \{\mathcal{S} \cup \{0\}\}$ is the channel UE i uses at time slot t. Note that $\alpha_i(t) = 0$ indicates that no channel is assigned to UE i. In addition, each UE can choose from three transmission modes: *uplink*, *downlink*, or *full-duplex*. The transmission mode selection is denoted as $\beta_i(t) \in \{U,\ D,\ F\}$, where $\beta_i(t) = U$, $\beta_i(t) = D$, and $\beta_i(t) = F$ indicate that at time slot t, UE i selects half-duplex uplink, half-duplex downlink, and full-duplex transmission, respectively.

For the full-duplex mode, the residual self-interference is treated as interference. Let $C_i^u(t)|_{\alpha_i(t)=s,\beta_i(t)=F}$ and $C_i^d(t)|_{\alpha_i(t)=s,\beta_i(t)=F}$ be the uplink and downlink channel capacity of UE i at time slot t, respectively, given that channel s is assigned to UE i and the full-duplex mode is selected. We have

$$C_i^u(t)|_{\alpha_i(t)=s,\beta_i(t)=F} = B\log_2\left(1+\frac{p_i^u(t)|h_s^u|^2}{N_0+p_i^d\eta_d}\right) \tag{4.2}$$

$$C_i^d(t)|_{\alpha_i(t)=s,\beta_i(t)=F} = B\log_2\left(1+\frac{p_i^d(t)|h_s^d|^2}{N_0+p_i^u\eta_u}\right), \tag{4.3}$$

where B is the channel bandwidth; h_s^u and h_s^d are the channel gains between the AP and UE i for the uplink and downlink channel, respectively; $p_i^u(t) > 0$ and $p_i^d(t) > 0$ are the uplink and downlink transmit power, respectively; η_d and η_u are the self-interference cancellation ratio at the AP and a UE, respectively; and N_0 is additive white Gaussian noise power.

For half-duplex uplink transmission, the uplink channel capacity for UE i, given that it is assigned with channel s, is

$$C_i^u(t)|_{\alpha_i(t)=s,\beta_i(t)=U} = B\log_2\left(1+\frac{p_i^u(t)|h_s^u|^2}{N_0}\right). \tag{4.4}$$

In this case, we have $p_i^u(t) > 0$ and $p_i^d(t) = 0$. For half-duplex downlink transmission, the downlink channel capacity for UE i, given that it is assigned with channel s, is

$$C_i^d(t)|_{\alpha_i(t)=s,\beta_i(t)=D} = B\log_2\left(1+\frac{p_i^d(t)|h_s^d|^2}{N_0}\right). \tag{4.5}$$

In this case, we have $p_i^u(t) = 0$ and $p_i^d(t) > 0$.

The dynamics of the uplink and downlink queues can be written as

$$Q_i^u(t+1) = \max\{Q_i^u(t) + A_i^u(t) - B_i^u(t), 0\} \tag{4.6}$$

$$Q_i^d(t+1) = \max\{Q_i^d(t) + A_i^d(t) - B_i^d(t), 0\}, \tag{4.7}$$

where $B_i^u(t) = \frac{T}{L} C_i^u(t)|_{\alpha_i(t),\, \beta_i(t)}$ and $B_i^d(t) = \frac{T}{L} C_i^d(t)|_{\alpha_i(t),\, \beta_i(t)}$ are the service rates in packets per time slot at time t for the uplink and downlink queues, respectively, T is the duration of a time slot, and L is the packet length in bits.

4.2.2 Problem formulation

As can be seen from (4.2)–(4.5), the overall throughput can be enhanced with full-duplex transmissions, but at the cost of higher energy consumption. The energy efficiency maybe degraded due to the residual self-interference. There is a trade-off between the overall queue length (which is indicative of delay) and energy efficiency with different transmission mode selections. Furthermore, both energy efficiency and throughput can be enhanced by transmitting only on good channels. However, there may be the extra delay to wait for the channel condition to be good from a deep fade.

The average total energy consumption of the system can be written as

$$\bar{P} \triangleq \limsup_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T-1} \sum_{i=1}^{N} \mathbb{E}\{p_i^u(t) + p_i^d(t) | \alpha_i(t), \beta_i(t)\} \tag{4.8}$$

We also define the average queue length as $\bar{Q} \triangleq \limsup_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T-1} \sum_{i=1}^{N} \mathbb{E}\{Q_i^u(t) + Q_i^d(t)\}$. We schedule the uplink and downlink transmissions at the beginning of each time slot. According to the notion of *throughput-optimal* [45], the objective is to minimize the average energy consumption while keeping all the uplink and downlink queues stable. We have the following problem formulation.

$$\min : \bar{P} = \limsup_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T-1} \sum_{i=1}^{N} \mathbb{E}\{p_i^u(t) + p_i^d(t) | \alpha_i(t), \beta_i(t)\} \tag{4.9}$$

$$\text{s.t. } \alpha_i(t) \neq \alpha_j(t), \text{if } \alpha_i(t) \in \mathcal{S} \text{ or } \alpha_j(t) \in \mathcal{S}, \text{ for all } i \neq j, i, j \in \mathcal{N} \tag{4.10}$$

$$\bar{Q} < \infty, \text{ for all } \{\vec{\lambda}^u, \vec{\lambda}^d\} \in \vec{\Lambda}, \tag{4.11}$$

where $\vec{\Lambda}$ is the capacity region of the WLAN system. Constraint (4.10) forbids two nodes accessing the same channel and Constraint (4.11) ensures that the schedule meets the notion of throughput-optimal.

4.3 Solution Algorithm and Performance Analysis

4.3.1 Lyapunov Optimization Based Scheduling Algorithm

Following the Lyapunov optimization framework, we first define the Lyapunov function $L(Q(t))$ as $L(Q(t)) \triangleq \frac{1}{2}\sum_{i=1}^{N}\left\{\{Q_i^u(t)\}^2 + \{Q_i^d(t)\}^2\right\}$, where $L(Q(0)) = 0$. Note that $L(Q(t))$ is small if and only if all the queue lengths are small; $L(Q(t))$ will become large if any of the queues is congested. The system is thus stable when $\mathbb{E}\{L(Q(t))\} < \infty$.

We then define the drift $\Delta(L(t))$ as

$$\Delta(L(t)) \triangleq \mathbb{E}\{L(Q(t+1)) - L(Q(t))\,|Q(t)\}. \tag{4.12}$$

The system is stable when

$$\begin{aligned}\mathbb{E}\{L(Q(t))\} =& \mathbb{E}\left\{\sum_{k=0}^{t-1}[L(Q(k+1)) - L(Q(k))]\right\} \\ =& \sum_{k=0}^{t-1}\mathbb{E}\{L(Q(k+1)) - L(Q(k))|Q(k)\} \\ =& \sum_{k=0}^{t-1}\Delta(L(k)) < \infty\end{aligned} \tag{4.13}$$

We can minimize the drift in every time slot t to maintain a finite expectation for $L(Q(t))$.

It follows the queue dynamics (4.6) and (4.7) that

$$\begin{aligned}&\{Q_i^u(t+1)\}^2 + \{Q_i^d(t+1)\}^2 \\ \leq& \{Q_i^u(t) + A_i^u(t) - B_i^u(t)\}^2 + \{Q_i^d(t) + A_i^d(t) - B_i^d(t)\}^2\end{aligned}$$

$$= \{Q_i^u(t)\}^2 + \{A_i^u(t) - B_i^u(t)\}^2 + 2Q_i^u(t)(A_i^u(t) - B_i^u(t)) +$$
$$\{Q_i^d(t)\}^2 + \{A_i^d(t) - B_i^d(t)\}^2 + 2Q_i^d(t)(A_i^d(t) - B_i^d(t)). \quad (4.14)$$

Substituting (4.14) into (4.12), we have

$$\Delta(L(t)) \leq \Phi + \mathbb{E}\left\{\sum_{i=1}^{N}\{Q_i^u(t)(A_i^u(t) - B_i^u(t)) + Q_i^d(t)(A_i^d(t) - B_i^d(t))\}\right\}, \quad (4.15)$$

where $\Phi = \frac{1}{2}\mathbb{E}\left\{\sum_{i=1}^{N}\{[A_i^u(t) - B_i^u(t)]^2 + [A_i^d(t) - B_i^d(t)]^2\}\right\}$, which is bounded if the arrival rate and service rate of each uplink and downlink queue are bounded. This is true if the arrival rates are within the capacity region of the system.

Defining $P(t) \triangleq \sum_{i=1}^{N}\left\{p_i^u(t) + p_i^d(t)\right\}$, we then obtain the *drift-plus-penalty* $\Delta(L(t)) + V\mathbb{E}\{P(t)\}$ as in [9], by incorporating the energy penalty (i.e., the overall energy consumption at time t) with a positive coefficient V. Parameter V indicates the UEs' emphasis on energy consumption. That is, the more emphasis on energy consumption, the greater the value of V. In particular, $V = 0$ indicates that the UEs are not sensitive to energy consumption at all. Based on (4.15), we can derive an upper bound on the *drift-plus-penalty* as

$$\Delta(L(t)) + V\mathbb{E}\{P(t)\}$$
$$\leq \Phi + \mathbb{E}\left\{\sum_{i=1}^{N}\{Q_i^u(t)(A_i^u(t) - B_i^u(t)) + Q_i^d(t)(A_i^d(t) - B_i^d(t))\} + VP(t)\right\}.$$

We minimize the second term on the right-hand-side $\Theta \triangleq \sum_{i=1}^{N}\{Q_i^u(t)(A_i^u(t) - B_i^u(t)) + Q_i^d(t)(A_i^d(t) - B_i^d(t))\} + VP(t)$ at each time slot t in order to minimize the *drift-plus-penalty*. Notice that Θ can be rewritten as $\Theta = \sum_{i=1}^{N}\{Q_i^u(t)A_i^u(t) + Q_i^d(t)A_i^d(t)\} - \sum_{i=1}^{N}\{Q_i^u(t)B_i^u(t) - Vp_i^u(t) + Q_i^d(t)B_i^d(t) - Vp_i^d(t)\}$. Then first term on the right-hand-side, $\sum_{i=1}^{N}\{Q_i^u(t)A_i^u(t) + Q_i^d(t)A_i^d(t)\}$, only depends

on the arrival rates and the current queue lengths. Therefore, it doesn't affect the scheduling decision. We only need to maximize the second term of Θ, which is a function of both $\alpha_i(t)$ and $\beta_i(t)$.

Let the channel assignment be $\vec{\alpha}(t) = \{\alpha_1(t), \alpha_2(t), ..., \alpha_N(t)\}$ and the transmission mode selection be $\vec{\beta}(t) = \{\beta_1(t), \beta_2(t), ..., \beta_N(t)\}$. We have

$$\begin{aligned}\Psi(t)|_{\vec{\alpha}(t),\vec{\beta}(t)} &\triangleq \sum_{i=1}^{N}\{Q_i^u(t)B_i^u(t) - Vp_i^u(t) + Q_i^d(t)B_i^d(t) - Vp_i^d(t)\}|_{\alpha_i(t),\beta_i(t)} \\ &= \sum_{i=1}^{N} \psi_i(t)|_{\alpha_i(t),\beta_i(t)}, \end{aligned} \tag{4.16}$$

where $\psi_i(t)|_{\alpha_i(t),\beta_i(t)} = \{Q_i^u(t)B_i^u(t) - Vp_i^u(t) + Q_i^d(t)B_i^d(t) - Vp_i^d(t)\}|_{\alpha_i(t),\beta_i(t)}$. Let the optimal channel assignment be $\vec{\alpha}^*(t) = \{\alpha_1^*(t), \alpha_2^*(t), ..., \alpha_N^*(t)\}$ and the optimal transmission mode selection be $\vec{\beta}^*(t) = \{\beta_1^*(t), \beta_2^*(t), ..., \beta_N^*(t)\}$. To find the optimal schedule $\{\vec{\alpha}^*(t), \vec{\beta}^*(t)\}$, we first need to identify the transmission mode for a given channel assignment $\alpha_i(t) = s$ for each UE i. That is,

$$\beta_i^*(t)|_{\alpha_i(t)=s} = \underset{\beta_i(t)\in\{U,D,F\}}{\arg\max} \{\psi_i(t)|_{\alpha_i(t)=s,\beta_i(t)}\}. \tag{4.17}$$

Note that $\psi_i(t) = 0$ if no transmission is conducted. Therefore we have

$$\psi_i^*(t)|_{\alpha_i(t)=s} = \max\{\psi_i(t)|_{\alpha_i(t)=s,\beta_i^*(t)}, 0\} \tag{4.18}$$

$$\vec{\psi}_i^*(t)|_{\alpha_i(t)} \triangleq \{\psi_i^*(t)|_{\alpha_i(t)=1}, \psi_i^*(t)|_{\alpha_i(t)=2}, ..., \psi_i^*(t)|_{\alpha_i(t)=S}\}. \tag{4.19}$$

We need to find the maximum channel assignment $\vec{\alpha}^*(t)$ based on $\vec{\psi}_i^*(t)|_{\alpha_i(t)}$, for $i = 1, 2, ..., N$. The channel assignment problem can be transformed into a *maximum weighted bipartite matching problem*. In the bipartite graph $\mathcal{G}$, UEs and the channels represent the two independent sets of vertices: the set of UEs G_1 and the set of channels G_2. In graph $\mathcal{G}$, the weight of the edge between an vertex in G_1 (i.e., a UE i) and another vertex in G_2 (i.e., a channel s) is set to $\psi_i^*(t)|_{\alpha_i(t)=s}$. This way, the maximum weighted bipartite matching of graph $\mathcal{G}$ corresponds to the optimal channel

Algorithm 4: Scheduling Algorithm for Channel Assignment and Transmission Mode Selection

Update all uplink and downlink queues and estimate all channel conditions at the beginning of each time slot t ;
For each UE i, find the transmission mode $\beta_i^*(t)|_{\alpha_i(t)=s}$ as in (4.17) ;
Obtain the channel assignment matrix $\{\vec{\psi}_1^*(t)|_{\alpha_1(t)}^T, \vec{\psi}_2^*(t)|_{\alpha_2(t)}^T, ..., \vec{\psi}_N^*(t)|_{\alpha_N(t)}^T\}$;
Apply the Hungarian Method and (4.17) to find the optimal schedule $\{\vec{\alpha}^*(t), \vec{\beta}^*(t)\}$;
if $\psi_i(t)|_{\{\alpha_i^*(t),\beta_i^*(t)\}} > 0$ **then**
    UE i transmits on channel $\alpha_i^*(t)$ with transmission mode $\beta_i^*(t)$;
end

assignment $\vec{\alpha}^*(t)$. The maximum weighted bipartite matching problem can be solved with the Hungarian Method [47]. The complexity of the Hungarian Method is $O(NS^2)$ if $N > S$, or $O(N^2S)$ if $N \leq S$.

When the optimal channel assignment is derived, the optimal transmission mode $\beta_i^*(t)$ for UE i is readily obtained as in (4.17), i.e., $\beta_i^*(t) = \beta_i^*(t)|_{\alpha_i^*(t)}$. Now we obtain the optimal schedule $\{\vec{\alpha}^*(t), \vec{\beta}^*(t)\}$ as well as the corresponding $\Psi(t)|_{\vec{\alpha}^*(t),\vec{\beta}^*(t)}$. Then we can assign the channels and decide the transmission mode for each UE based on the optimal schedule. Note that $\psi_i(t)|_{\alpha_i^*(t),\beta_i^*(t)} = 0$ if no transmission is scheduled for UE i; so UE i transmits if and only if $\psi_i(t)|_{\alpha_i^*(t),\beta_i^*(t)} > 0$.

The detailed algorithm for deriving the optimum schedule $\{\vec{\alpha}^*(t), \vec{\beta}^*(t)\}$ is presented in Algorithm 4, which is executed at the beginning of each time slot.

4.3.2 Performance Analysis

We have the following theorems for the performance of Algorithm 4. The proofs are omitted for lack of space.

Theorem 4.1. *The schedule $\{\vec{\alpha}^*(t), \vec{\beta}^*(t)\}$ obtained by Algorithm 4 achieves the maximum $\Psi(t)$.*

We also derive the upper bounds for the expectations of average sum queue lengths of all the uplink and downlink queues and the corresponding average total energy consumption as follows.

Theorem 4.2. *Assume that the arrival rates to the queues $\vec{\lambda}^u$ and $\vec{\lambda}^d$ are strictly within the system's capacity region, i.e., the system can be stabilized under certain $\{\vec{\alpha}(t), \vec{\beta}(t)\}$. Then the upper bounds on the average sum queue lengths and average energy consumption under Algorithm 4 can be derived as*

$$\limsup_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T-1} \sum_{i=1}^{N} \mathbb{E}\{Q_i^u(t) + Q_i^d(t)\} \leq \frac{1}{\epsilon}(\Phi + V\bar{P}) \tag{4.20}$$

$$\limsup_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T-1} \sum_{i=1}^{N} \mathbb{E}\{p_i^u(t) + p_i^d(t)\} \leq \bar{P}^{opt} + \frac{\Phi}{V}, \tag{4.21}$$

where $\bar{P}^{opt}$ is the minimum average energy consumption under any stable scheduling strategy, $\bar{P}$ is the average energy consumption under the proposed algorithm, $\epsilon > 0$ is the distance between the arrival rates $\{\vec{\lambda}^u, \vec{\lambda}^d\}$ and the system capacity region under the proposed algorithm, and Φ is given in (4.15).

4.4 Performance Evaluation

In this section, we evaluate the performance of the proposed algorithm through Matlab simulations. We assume that the maximum transmit power is 46 dBm at the AP and 23 dBm at the UEs. We assume that there is a 110 dB self-interference cancellation in both the uplink and downlink transceivers. For the wireless channels, we adopt the commonly used Okumura-Hata model for small and medium-sized cities. Each channel has a bandwidth of 360kHz. We assume that there are 12 UEs and 10 channels in the WLAN.

We compare the average energy consumptions and queue lengths of a half-duplex only system and a full-duplex system under different V values. The simulation results are presented in Figs. 4.1 and 4.2 for different traffic arrival rates. From the simulations, we find that the full-duplex system always outperforms the half-duplex only system with respect to both average queue length and energy consumption. Moreover, there is a trade-off between the average queue length and energy consumption for the full-duplex system under different V values.

Fig. 4.1 presents the average queue length versus traffic load. When $V = 0$, the scheme only minimizes the drift and does not care about energy consumption. In this case, the average queue length of the half-duplex case is always greater than that of the full-duplex case. Moreover, in the half-duplex only case, the queues cannot be stabilized when the arrival rate exceeds 25. In the full-duplex case, the queues can be stabilized until the arrival rate reaches 38. Clearly, full-duplex transmissions are helpful to keep the queue backlog low and increase the capacity region of the WLAN. It is also interesting to see that for all the full-duplex cases, the queues can be stabilized when the arrival rate is lower than 38, indicating that different V values do not affect the stability of the system. Moreover, the average queue length increases when V is increased, as indicated by the upper bound of average queue length (4.20) in Theorem 4.2.

Fig. 4.2 presents the average energy consumption versus traffic load. We find the average energy consumption of the half-duplex only case is smaller than that of the full-duplex cases under heavy load, when the queues become unstable. However, in the stable capacity region of the half-duplex only case (i.e., when the arrival rate is lower than 25), the average energy consumption of the half-duplex only case is greater than that of the full-duplex cases with $V > 50$. This is because when $V > 50$, the energy consumption is more seriously considered (i.e., in the drift-plus-penalty) and the UEs would transmit only when the energy efficiency is high. For the full-duplex case with $V = 0$, the average energy consumption is the highest among all the cases, since the proposed scheme does not consider energy efficiency. Furthermore, the energy consumption drops when the arrival rate is greater than 38. This is due to the unbalanced service rates of the uplink and downlink. When the queues are not stable, more uplink transmissions were made; the uplink transmit power is comparatively smaller than that of the downlink transmissions. Finally, it can be seen that the energy consumption decreases when V is increased, as indicated by the upper bound of average energy consumption (4.21) in Theorem 4.2.

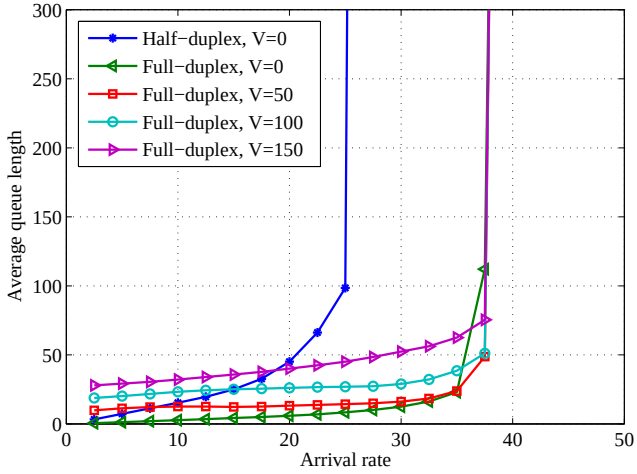

Figure 4.1: Average queue lengths achieved by the proposed algorithm: half-duplex only with V=0, full-duplex with V=0, full-duplex with V=50, full-duplex with V=100, and full-duplex with V=150.

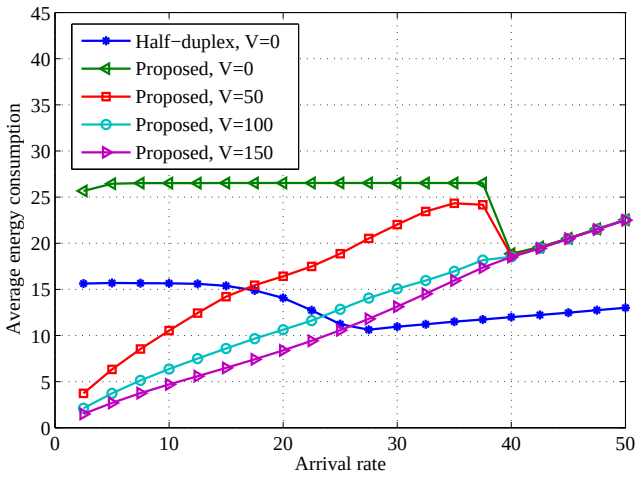

Figure 4.2: Average energy consumptions achieved by the proposed algorithm: half-duplex with V=0, full-duplex with V=0, full-duplex with V=50, full-duplex with V=100, and full-duplex with V=150.

4.5 Conclusion

In this chapter, we proposed an online scheduling algorithm to jointly decide the channel assignment, transmission scheduling, half- or full-duplex transmission mode selection for each UE in a multi-channel ful-duplex WLAN. The proposed scheme was based on Lyapunov optimization. We also proved the optimality of the proposed algorithm and derived upper bounds for the average

queue length and energy consumption under the proposed algorithm. We evaluated the performance of the proposed algorithm with simulations. We showed that under the proposed algorithm, there was a trade-off between the average queue length and energy consumption under different V values.

4.6 Appendix

4.6.1 Proof For Theorem.4.2

According to Theorem.4.2, Algorithm 4 maximizes $\Psi(t)$, which minimizes $\Psi(t)$. And we have

$$\begin{aligned}
&\min\{\Psi(t)\} \qquad (4.22)\\
=&\min\{\sum_{i=1}^{N}\{Q_i^u(t)(A_i^u(t)-B_i^u(t))+Q_i^d(t)(A_i^d(t)-B_i^d(t))\}\\
&+VP(t)\}\\
\leq&\sum_{i=1}^{N}\{Q_i^{u*}(t)(A_i^u(t)-B_i^{u*}(t))+Q_i^{d*}(t)(A_i^d(t)-B_i^{d*}(t))\}\\
&+VP^*(t)
\end{aligned}$$

Where Q_i^{u*}, Q_i^{d*}, $B_i^{d*}(t)$ and $P^*(t)$ are the terms corresponding to any (possible randomized) scheme. Now consider a randomized scheduling policy that achieves the optimal energy consumption and stabilizes the system, i.e., for $i \in \mathcal{N}$

$$\mathbb{E}\{P^*(t)\} = P^{opt} \qquad (4.23)$$

$$\mathbb{E}\{A_i^u(t) - B_i^{u*}(t)\} \leq 0 \qquad (4.24)$$

$$\mathbb{E}\{A_i^d(t) - B_i^{d*}(t)\} \leq 0 \qquad (4.25)$$

where p^{opt} is the minimal energy consumption under the corresponding scheduling policy that stabilizes the system, ie, (4.24) and (4.25) is fulfilled.

Then under algorithm 4, we have

$$\begin{aligned}
&\Delta(L(t)) + \{P(t)\} \qquad (4.26)\\
\leq &\Phi + \mathbb{E}\{\min\{\Psi(t)\}\}\\
\leq &\Phi + \mathbb{E}\left\{\sum_{i=1}^{N}\{Q_i^{u*}(t)(A_i^u(t) - B_i^{u*}(t))\}\right\}\\
&+ \mathbb{E}\left\{\sum_{i=1}^{N}\{Q_i^{d*}(t)(A_i^d(t) - B_i^{d*}(t))\}\right\} + V\mathbb{E}\{P^*\}
\end{aligned}$$

As $Q_i^{u*}(t)$ is independent to $(A_i^u(t) - B_i^{u*}(t))$ and $Q_i^{d*}(t)$ is independent to $(A_i^d(t) - B_i^{d*}(t))$, along with (4.24) and (4.25), we have

$$\mathbb{E}\{Q_i^{u*}(t)(A_i^u(t) - B_i^{u*}(t))\} \leq 0 \qquad (4.27)$$

$$\mathbb{E}\{Q_i^{d*}(t)(A_i^d(t) - B_i^{d*}(t))\} \leq 0 \qquad (4.28)$$

for all $i \in \mathcal{N}$. Then substitute (4.27) and (4.28) to (4.22), we have

$$\begin{aligned}
&\Delta(L(t)) + \mathbb{E}\{P(t)\} \qquad (4.29)\\
\leq &\Phi + 0 + V\mathbb{E}\{P^*\} = \Phi + P^{opt}
\end{aligned}$$

For a stable system, we have $\sum_{k=1}^{T-1}\Delta(L(t)) = L(t) < \infty$, it follows that

$$\begin{aligned}
&\limsup_{t\to\infty}\frac{1}{T}\sum_{k=0}^{T-1}\Delta(L(t)) + \limsup_{t\to\infty}\frac{V_p}{T}\sum_{k=0}^{T-1}\mathbb{E}\{P(t)\} \qquad (4.30)\\
= &\limsup_{t\to\infty}\frac{V_p}{T}\sum_{k=0}^{T-1}\mathbb{E}\{P(t)\}\\
= &\limsup_{t\to\infty}\frac{V_p}{T}\sum_{k=0}^{T-1}\mathbb{E}\{p_i^u(t) + p_i^d(t)\}
\end{aligned}$$

$$\leq \Phi + V_p \cdot P^{opt}$$

Then (4.21) hold true.

Suppose that the system is can be stabilized under the proposed scheduling algorithm and for all $i \in \mathcal{N}$ there exist a real number $\epsilon \geq 0$, such that

$$\mathbb{E}\{A_i^u(t) - B_i^u(t) \leq -\epsilon\} \tag{4.31}$$

$$\mathbb{E}\{A_i^d(t) - B_i^d(t) \leq -\epsilon\} \tag{4.32}$$

then we have

$$\begin{aligned}
&\Delta(L(t)) + \mathbb{E}\{P(t)\} \qquad (4.33)\\
\leq &\Phi + \mathbb{E}\left\{\sum_{i=1}^{N}\{Q_i^u(t)(A_i^u(t) - B_i^u(t))\}\right\}\\
&+ \mathbb{E}\left\{\sum_{i=1}^{N}\{Q_i^d(t)(A_i^d(t) - B_i^d(t))\}\right\} + V\mathbb{E}\{P(t)\}\\
\leq &\Phi - \mathbb{E}\left\{\sum_{i=1}^{N}\epsilon Q_i^u(t)\right\}\\
&- \mathbb{E}\left\{\sum_{i=1}^{N}\epsilon Q_i^d(t)\right\} + V\mathbb{E}\{P(t)\}\\
= &\Phi - \epsilon\mathbb{E}\left\{\sum_{i}^{N}\{Q_i^u(t) + Q_i^d(t)\}\right\} + V\mathbb{E}\{P(t)\}
\end{aligned}$$

Note that $\Delta(L(t)) + \mathbb{E}\{P(t)\} \geq 0$ as is guaranteed in Algorithm 4, we have

$$\mathbb{E}\left\{\sum_{i}^{N}\{Q_i^u(t) + Q_i^d(t)\}\right\} \leq \frac{1}{\epsilon}\{\Phi + V\mathbb{E}\{P(t)\}\} \tag{4.34}$$

It follows that

$$\begin{aligned}
&\limsup_{t\to\infty}\frac{1}{T}\sum_{t=1}^{T-1}\sum_{i=1}^{N}\mathbb{E}\{Q_i^u(t)+Q_i^d(t)\} \qquad (4.35)\\
\leq&\frac{\Phi}{\epsilon}+\frac{1}{\epsilon}\limsup_{t\to\infty}\frac{1}{T}\{V_p\mathbb{E}\{P(t)\}\}\\
=&\frac{1}{\epsilon}(\Phi+V_pP)
\end{aligned}$$

Then (4.21) holds.

Chapter 5

Access Strategy and Dynamic Downlink Resource Allocation for Femtocell Networks

5.1 Introduction

Femtocells, also named as Femto Access Points (FAP), are small, low power cellular base stations (BS). Femtocells are designed for use at homes and small enterprises, and are usually connected to the core network with broadband wireline connections [50]. In addition to providing a shortcut to the core network, the wireline connection also enables coordinations among FAPs and macrocell base stations (MBS) to improve the performance of the two-tier network. Femtocells are considered as a low-cost and effective solution to extend wireless coverage and offload voice and wireless data. This is really important, as research indicates that 70% of data traffic take place indoor where the coverage of conventional cellular networks is usually poor. With femtocells, the distance between BS and a User Equipments (UE) is greatly reduced, thus enabling better signal transmissions and better spatial reuse of spectrum.

The success of femtocell networks largely relies on the management of interference. The deployment of femtocells provides better coverage to nearby Femtocell User Equipments (FUE), but it may also produce a “dead zone” to nearby Macro User Equipments (MUE). FAPs are usually deployed in places where there is poor MBS coverage; the MUE and MBS must use high transmit power to sustain their connection, thus leading to strong interference to FUEs. Unlike well-planned and optimized deployment of cellular networks (i.e., the MBS’s), FAPs are usually installed by end-users in a chaotic manner. The coverage of FAPs may overlap with each other and cause interference among FAPs themselves.

From the perspective of access policy, femtocells can be classified into (i) closed access, where only subscribers can access the FAPs, and (ii) open access, where an FAP serves both subscribers and nearby MUEs. Although open access is more appealing for interference management, its success depends on the willingness of the FAPs to serve non-subscribed MUEs; some incentive mechanisms would be critical to encourage FAP owners to adopt this strategy. From the perspective of spectrum resource allocation, femtocells can be classified into (i) co-channel scenarios, where MBS's and FAPs share the spectrum band, and (ii) dedicated channel scenarios, where orthogonal channels are assigned. The tension between interference and spectrum efficiency should be carefully balanced.

In this work, we investigate the problem of access control and spectrum resource allocation in two-tier femtocell networks. We assume one MBS and multiple FAPs in the area and consider the open access scheme. The FUEs are always connected to the corresponding FAPs, while the MUEs can choose between the MBS and a nearby FAP for connection. The spectrum is divided into two parts, one for the MBS and the other part for the FAPs. To provide incentives to FAPs for serving MUEs, we allow dynamic partition of the spectrum according to the network dynamics; more bandwidth will be allocated to the FAPs if they serve more MUEs.

We developed a scheme for joint access control and spectrum resource allocation. The goal is to maximize the network-wide capacity and improve the performance of UEs with poor MBS coverage, by assigning the MUEs to the MBS or FAPs and by dynamically partition the spectrum for the MBS and the FAPs. We also aim to guarantee the quality of service (QoS) of the users in the form of a minimum capacity requirement. The formulated problem is a mixed integer nonlinear programming (MINLP) problem. We then develop an algorithm that assigns MUEs to the BS's and an algorithm for allocating spectrum resource to the BS's once the BS association for the MUEs are determined. An upper bound on the network capacity achieved by the proposed algorithms is also derived. The performance of the proposed algorithms are evaluated with simulations, and are shown to outperform an existing scheme with considerable gains. The upper bound is also found to be quite tight for most of the cases examined in the simulation study.

The remainder of the chapter is organized as follows. We discuss related work in Section 5.2 and the problem formulation in Section 5.3. In Section 5.4, we propose access control and spectrum resource allocation algorithms and derive the capacity upper bound. Simulation studies are presented in Section 5.6. Section 5.7 concludes the chapter.

5.2 Related Work

Compared with Wi-Fi access points, femtocells provide a solution of supporting better voice and data coverage by switching from the cellular network to another service provider when the signal quality is poor indoor, instead of just providing high speed data transmissions. Femtocells are now primarily viewed as a cost-effective means of offloading data and voice from the macrocell network [50]. Because of the advantages for both network operators and customers, the benefits of femtocells cannot be overemphasized in the long term. However, the two-tier architecture of macrocells and femtocells inevitably brings about the cross-tier interference problem. Further, as femtocells are usually deployed by end-users and the deployment of femtocells are not well planned, femtocells may be overlapped with each other, causing co-tier interference among such femtocells [58]. Hence, interference management in femtocell network has received tremendous attention from either academic or industrial areas [60, 61].

As the interference in femtocell network is largely determined by the deployment scenarios, Mahmoud and Guvenc in [54] summarized femtocell deployment from two perspectives: (i) closed access or open access, (ii) co-channel or dedicated channels. A game-theoretic approach for resource allocation in OFDMA femtocells with closed access was proposed in [55]. However, an non-subscribed user that is close to an FAP may be far away from the MBS. Its transmit power should be increased to meet its QoS requirement, thus introducing stronger interferences to users of the FAP. In [73], a self-optimized coverage coordination scheme was proposed to provide better indoor femtocell coverage and avoid leaking the femtocell coverage into an outdoor macrocell. In [52], the authors introduced a game-theoretic framework for the FAPs to decide their own access policy in order to maximize the system performance. And another game-theoretic approach in [77]

was proposed to mitigate the interference between Macrocell and femtocell. In [53], an algorithm was proposed for the open access scenario to improve network throughput, while a hybrid access mechanism was introduced in [56] to guarantee the resources for users and reduce interference. In [67, 68], neighborhood femtocell Handover schemes were developed improve the system performance in dense femtocellular network. In [70], a framework of Spectrum-Sharing Rewarding was proposed for hybrid access mechanism to maximize the benefit of femtocell owners. The performance of two-tier femtocell networks with partially open channels was evaluated in [71]. In co-channel scenarios, the spectrum is available for all users but it may lead to high cross-tier interference. To mitigate the interference in co-channel scenarios, a Frequency ALOHA (F-ALOHA) was adopted to avoid excessive cross-tier interference in [51]. In [64–66, 79, 80], some power adaptive schemes were developed to mitigate the interference. In [57, 60, 74], the authors proposed a Cognitive Radio (CR) approach to mitigate the cross-tier interference. In [49], the impact of Interference Alignment (IA) in femtocell networks was evaluated, and in [69], a game theory approach for IA was proposed. In [63], a interference avoidance strategy is developed in a two-tier CDMA network to mitigate the uplink interference. In [72], a resource allocation scheme with QoS constraints was proposed for the interference avoidance application. In [76], a joint subchannel scheme as well as a disjoint subchannel scheme were proposed for resource allocation in the two-tier femtocell network. The performance of two-tier femtocell networks with cochannel femtocell deployment was analysed with outage constraints in [78]. A Femtocell Identification (FID) approach was proposed in [75] to avoid co-channel interference between neighbour femtocells. In Co-channel deployment scenarios, it is usually difficult to guarantee the Quality of Service (QoS) requirements for users. In dedicated channel scenarios, spectrum is divided into orthogonal portions and allocated to different tiers, in order to eliminate cross-tier interference at the price of a lower spectrum efficiency [54].

5.3 System Model and Problem Statement

5.3.1 System Model

We consider a femtocell network with one MBS (indexed with 0) collocated with $\mathcal{N} = \{1, 2, \cdots, N\}$ FAPs. Let $\mathcal{L}_0 = \{1, 2, \cdots, L_0\}$ denote the set of active MUEs in the network. Each FAP $i \in \mathcal{N}$ serves a set of active FUEs, denoted as $\mathcal{L}_i = \{1, \ldots, L_i\}$, for $i = 1, 2, \cdots, N$.

The spectrum B for this femtocell network is divided into two parts: (i) B_0 allocated to the MBS, and (ii) the remaining portion $(B - B_0)$ allocated to the FAPs. An FAP i will use spectrum $(B - B_0)$ to serve its subscribers $\mathcal{L}_i$ and some of the MUEs; the remaining MUEs will be served by the MBS using spectrum B_0. Since the spectrum allocated to the MBS and the FAPs are orthogonal, there is no cross-tier interference.

Due to the autonomous, chaotic deployment of the FAPs, the set of FAPs can be classified into disjointed clusters. The FAPs in a cluster has overlapped coverage and may interfere with each other, but there is no interference among different clusters. If a cluster consists of an isolated FAP, the FAP can use all the $(B - B_0)$ spectrum without interfering other FAPs or the MBS. A cluster with multiple FAPs is treated as a "virtual" FAP. From the perspective of MUEs and the MBS, the cluster behaves like one FAP. Within the cluster, we assume the interfering FAPs are allocated with orthogonal spectrum resources in the $(B - B_0)$ band to avoid interference. For example, interference graphs can be used to model the exclusive relationship among the interfering FAPs [60].

In this work, we consider an open access scheme, in which all the MUEs are allowed to access a nearby FAP, while the FUEs always connect to the corresponding FAPs. Recall that $\mathcal{L}_k$ is the set of UEs subscribed to BS k, for $k = 0, 1, \cdots, N$ ($\mathcal{L}_0$ is the set of MUEs). For open access of the MUEs, we define a variable $\rho_{i,j}(k)$ to indicate the access strategy of a UE $j \in \mathcal{L}_k$ originally

subscribed to BS k.

$$\rho_{i,j}(k) = \begin{cases} 1, & \text{UE } j \in \mathcal{L}_k \text{ accesses BS } i \\ 0, & \text{otherwise,} \end{cases} \quad \forall\, i,k \in \{0\} \cup \mathcal{N}. \tag{5.1}$$

Since we assume that all FUEs in $\mathcal{L}_k$ access to the correspondent FAP k, it follows that $\rho_{k,j}(k) = 1$, for all $k \neq 0$.

As FAPs are usually deployed by customers for home or office use, we adopt the standard indoor propagation model for the FAP link between UE j, $j \in \mathcal{L}_k$, and BS i as [59]

$$\lambda_{i,j}(k) = 37 + 30\log_{10} d_{i,j}(k) + 18.3 n^{\left(\frac{n+2}{n+1} - 0.46\right)}, \quad \forall\, i,k \neq 0, \tag{5.2}$$

where $d_{i,j}(k)$ is the separation from BS i to UE j, for all $j \in \mathcal{L}_k$; n is the number of floors along the path. For the MBS, we adopt the standard outdoor model for the path loss from the MBS to MUE $j \in \mathcal{L}_0$ as [59]

$$\lambda_{0,j}(0) = 40\log_{10} d_{0,j}(0) + 30\log_{10} f + 49, \tag{5.3}$$

where f (in MHz) is the central carrier frequency. As the bandwidth of the spectrum is much small comparing to the carrier frequency, we can fix f to a constant f_0 for simplification.

Consider an addictive white Gaussian noise (AWGN) channel, the Signal to Interference plus Noise (SINR) of user j, $j \in \mathcal{L}_k$, from BS i is denoted as

$$\varepsilon_{i,j}(k) = p_{i,j}(k) h_{i,j}(k), j \in \mathcal{L}_k, i,k \in \{0,1,\cdots,N\}, \tag{5.4}$$

where $h_{i,j}(k) = 10^{(-\lambda_{i,j}(k)/10)}/(N_0 + I_j(k))$; $p_{i,j}(k)$ is the transmit power of BS i to UE j, $j \in \mathcal{L}_k$; N_0 denotes the power of background white Gaussian noise; $I_j(k)$ is the received interference of UE j, $j \in \mathcal{L}_k$, from nearby FAPs. Therefore, the downlink capacity for UE j, $j \in \mathcal{L}_k$ can be approximated by the Shannon capacity as

$$C_j(k) = \sum_{i=0}^{N} \rho_{i,j}(k) B_{i,j}(k) \log_2(1 + \varepsilon_{i,j}(k)), j \in \mathcal{L}_k, \forall\ k, \tag{5.5}$$

where $B_{i,j}(k)$ denotes the spectrum band allocated to UE j, $j \in \mathcal{L}_k$ by BS i. Then, the downlink capacity of BS i can be computed as

$$C_i = \sum_{k=0}^{N} \sum_{j \in \mathcal{L}_k} \rho_{i,j}(k) B_{i,j}(k) \log_2(1 + \varepsilon_{i,j}(k)), \forall\ i. \tag{5.6}$$

5.3.2 Problem Formulation

In femtocell networks, the deployment of FAPs makes the transmitter and receiver closer to each other, hence offering better QoS and reducing power consumption and interference. However, FAPs may introduce strong interference to, or be interfered by nearby MUEs, if the same spectrum is used. Consequently, some open access schemes have been introduced as a means for mitigating such cross-tier interference. However, it is usually hard to persuade FAP owners to offer open access to non-subscribed users, as FAPs are installed and owned by end-users, rather than service providers.

In this work, we propose an incentive scheme that compensates FAPs with spectrum resource for offering open access to nearby MUEs. Specifically, we dynamically partition the spectrum resource according to the association of the MUEs. If more MUEs are switched to nearby FAPs for better service, the MBS share of the spectrum B_0 will be reduced and more spectrum will be allocated to the FAPs. Since the FAP clusters are not interfering with each other, the share $(B - B_0)$ can be used by all the FAP clusters simultaneously, achieving the gain of spatial reuse. It is worth noting that the share $(B - B_0)$ for FAPs is determined by the FAP cluster that serves the most

MUEs. For other FAPs serving fewer MUEs, the extra spectrum can be allocated to their FUEs for better service, as an additional incentive for the FAPs to serve MUEs.

The objective is then to maximize the overall capacity of the femtocell network. To achieve this goal, an efficient access scheme for the MUEs and a corresponding spectrum allocate mechanism are needed to dynamically determine the spectrum partition and the spectrum resource allocated to each UE. The constraints are the total spectrum resource of the system and the QoS requirements of the UEs. The dynamic access and resource allocation problem can be formulated as follows.

$$\text{maximize} \quad \sum_{i=0}^{N} C_i \tag{5.7}$$

subject to:

$$\rho_{i,j}(k) \in \{0,1\}, \quad i,k \in \{0\} \cup \mathcal{N}, j \in \mathcal{L}_k \tag{5.8}$$

$$\rho_{k,j}(k) = 1, k \in \mathcal{N}, j \in \mathcal{L}_k \tag{5.9}$$

$$\sum_{i} \rho_{i,j}(k) = 1, i,k \in \{0\} \cup \mathcal{N}, j \in \mathcal{L}_k \tag{5.10}$$

$$B_{i,j}(k) \geq 0, i,k \in \{0\} \cup \mathcal{N}, j \in \mathcal{L}_k \tag{5.11}$$

$$\sum_{k} \sum_{j} B_{i,j}(k)\rho_{i,j}(k) = B_0, \quad i \in \{0\}, k \in \{0\} \cup \mathcal{N}, j \in \mathcal{L}_k \tag{5.12}$$

$$\sum_{k} \sum_{j} B_{i,j}(k)\rho_{i,j}(k) = B - B_0, \quad i \in \mathcal{N}, k \in \{0\} \cup \mathcal{N}, j \in \mathcal{L}_k \tag{5.13}$$

$$0 \leq C_j(k) \leq \mathcal{C}, k \in \{0\} \cup \mathcal{N}, j \in \mathcal{L}_k. \tag{5.14}$$

In the formulated problem, constraint (5.9) indicates that an FUE can only access the FAP to which it subscribes, while constraint (5.10) indicates that a UE can only access the MBS or one FAP at a time. Constraint (5.12) represents the fact that the MBS have spectrum resource B_0 for

all the MUEs, while constraint (5.13) represents the fact that the FAPs have spectrum resource $(B - B_0)$. Constraint (5.14) is the QoS requirement that the downlink capacity of each UE should be no less than $\mathcal{C}$.

We aim to maximize the capacity of the entire network. The solution of this problem involves optimizing the access strategy of the MUEs (i.e., determining the binary values of $\rho_{i,j}(k)$'s) and the allocation of the spectrum resource (i.e., determine the non-negative real values of $B_{i,j}(k)$'s). Problem (5.7) is an MINLP problem, which is NP-hard in general. In the following section, we proposed an algorithm to solve this problem with near-optimal solutions as well as a proven performance bound.

5.4 Algorithms and Performance Bound

In this section, we first reformulate problem (5.7) to obtain a simplified version. Based on observations obtained from the reformulation, we then develop two algorithms that assign the MUEs to either the MBS or an FAP based on the achievable capacity gains, and then to allocate the spectrum resource to the FAPs, We also develop an upper bound for the network-wide capacity achieved by the proposed algorithms.

5.4.1 Solution Algorithms

To solve the problem, we first simplify it by reformulating the objective function (5.7). Based on (5.8), (5.9) and (5.10), the objective function (5.7) can be reformulated as in (5.16). According to the reformulation in (5.16), the total capacity of the network can be divided into two parts:

- the capacity achieved by the MUEs served by the MBS, which shares a total spectrum of B_0 (see (5.12)).
- the capacity achieved by the MUEs served by FAPs and the capacity achieved by the FUEs, where each FAP cluster has spectrum resource of $(B - B_0)$ (see (5.13)).

$$
\begin{aligned}
&\sum_{i=0}^{N} C_i = C_0 + \sum_{i=1}^{N} C_i \\
&= \sum_{k=0}^{N} \sum_{j \in \mathcal{L}_k} \rho_{0,j}(k) B_{0,j}(k) \log_2(1 + \varepsilon_{0,j}(k)) + \sum_{i=1}^{N} \sum_{k=0}^{N} \sum_{j \in \mathcal{L}_k} \rho_{i,j}(k) B_{i,j}(k) \log_2(1 + \varepsilon_{i,j}(k)) \\
&= \sum_{j \in \mathcal{L}_0} \rho_{0,j}(0) B_{0,j}(0) \log_2(1 + \varepsilon_{0,j}(0)) \\
&\quad + \sum_{i=1}^{N} \left\{ \sum_{j \in \mathcal{L}_0} \rho_{i,j}(0) B_{i,j}(0) \log_2(1 + \varepsilon_{i,j}(0)) + \sum_{j \in \mathcal{L}_i} \rho_{i,j}(i) B_{i,j}(i) \log_2(1 + \varepsilon_{i,j}(i)) \right\} \qquad (5.15) \\
&= \sum_{j \in \mathcal{L}_0} \sum_{i=0}^{N} \rho_{i,j}(0) B_{i,j}(0) \log_2(1 + \varepsilon_{i,j}(0)) + \sum_{i=1}^{N} \sum_{j \in \mathcal{L}_i} \rho_{i,j}(i) B_{i,j}(i) \log_2(1 + \varepsilon_{i,j}(i)). \qquad (5.16)
\end{aligned}
$$

According to (5.16), the first component to reformulate is the capacity achieved by MUEs. Let $B_{i,j}(0) \equiv B_j$, where B_j is a constant, for all base stations i and MUE $j \in \mathcal{L}_0$. That is, for MUE j it should be allocated with the same amount of spectrum resource no matter which base station it connects to. It follows that

$$
\begin{aligned}
&\sum_{j \in \mathcal{L}_0} \sum_{i=0}^{N} \rho_{i,j}(0) B_{i,j}(0) \log_2(1 + \varepsilon_{i,j}(0)) \\
&= \sum_{j \in \mathcal{L}_0} \sum_{i=0}^{N} \rho_{i,j}(0) B_j \log_2(1 + \varepsilon_{i,j}(0)) \\
&\leq \sum_{j \in \mathcal{L}_0} B_j \cdot \max_{\{0 \leq i \leq N\}} \{\log_2(1 + \varepsilon_{i,j}(0))\}. \qquad (5.17)
\end{aligned}
$$

The inequality is because there is only one $\rho_{i,j}(0)$ is one and all others are zero. Hence, each MUE should access an MBS or FAP that offers the best SINR for the downlink link.

Consider the case when $\max_{\{0 \leq i \leq N\}} \{\log_2(1 + \varepsilon_{i,j}(0))\} = \log_2(1 + \varepsilon_{0,j}(0))$, i.e., the MBS can offer the best SINR for MUE j. Even in this case, accessing a nearby FAP may still bring a larger capacity gain for the entire network, since the spectrum resource allocated to the FAPs can be spatially reused. Define

$$
G_{i,j}(k) = \log_2(1 + \varepsilon_{i,j}(k)), \qquad (5.18)
$$

and let ψ_i denote the maximum $G_{i,j}(k)$ among the UEs served by BS i, i.e.,

$$\psi_i = \max_{j \in \mathcal{L}_k, 0 \le k \le N} \{G_{i,j}(k)\}, \forall\, i, \tag{5.19}$$

and define n^* as

$$n^* = \arg\max_{1 \le i \le N} \{G_{i,j}(0)\}. \tag{5.20}$$

In this case, if the following condition is satisfied, i.e.,

$$G_{0,j}(0) < \sum_{i \in \mathcal{N} \setminus n^*} \psi_i - \psi_{n^*} \left(\frac{G_{0,j}(0)}{G_{n^*,j}(0)} - 1 \right) + G_{0,j}(0),$$

we have that

$$\sum_{i \in \mathcal{N} \setminus n^*} \psi_i \frac{1}{G_{0,j}(0)} - \psi_{n^*} \left(\frac{1}{G_{n^*,j}(0)} - \frac{1}{G_{0,j}(0)} \right) > 0. \tag{5.21}$$

Then an MUE can achieve larger network-wide capacity by accessing FAP n^*.

Theorem 5.1. *An MUE can achieve larger network-wide capacity by accessing the FAP n^*, which offers the best SINR among all FAPs, if the following inequality holds:*

$$\sum_{i \in \mathcal{N} \setminus n^*} \psi_i \frac{1}{G_{0,j}(0)} - \psi_{n^*} \left(\frac{1}{G_{n^*,j}(0)} - \frac{1}{G_{0,j}(0)} \right) > 0. \tag{5.22}$$

Proof. Consider that a MUE is now decided to access the MBS in the step one, and assume that it is allocated with spectrum bandwidth B. If it decide to access to FAP n^*, and assign the bandwidth B to FAPs, then it would bring about the change of capacity as following:

1. For MBS, capacity is decreased by $BG_{0,j}(0)$, as the MUE is served by a FAP other than the MBS and the corresponding spectrum bandwidth is assigned to FAPs.

2. For FAPs other than n^*, the capacity can be increased by $B \sum_{i \in \mathcal{N} \setminus n^*} \psi_i$ as the spectrum resource B is assigned to the FAPs.

3. For FAP n^*, the capacity would be increase by $-\psi_{n^*}\left(B\frac{G_{0,j}(0)}{G_{n^*,j}(0)}-1\right)+BG_{0,j}(0)$. As in step one, the MUE decided to access the MBS, then the SINR of the MBS must be better than that of any FAP. That is, $\frac{G_{0,j}(0)}{G_{n^*,j}(0)}>1$. So the FAP n^* need to assign a spectrum bandwidth $B\frac{G_{0,j}(0)}{G_{n^*,j}(0)}$ for the MUE to remain the same capacity. In this condition, the spectrum assign to the FUE served by FAP n^* with best SINR would decrease by $B*\frac{G_{0,j}(0)}{G_{n^*,j}(0)}-B$ and would bring about a capacity decrease of the $\psi_{n^*}B\left(\frac{G_{0,j}(0)}{G_{n^*,j}(0)}-1\right)$. In addition, as the MUE is served by FAP n^* and remain the same capacity, it brings about an capacity increase of $BG_{0,j}(0)$.

In summary, if the MUE decide to access FAP n^*, the network-wide capacity would increase by:

$$\Delta C=BG_{0,j}(0)+B\sum_{i\in\mathcal{N}\backslash n^*}\psi_i-\psi_{n^*}\left(B\frac{G_{0,j}(0)}{G_{n^*,j}(0)}-1\right)+BG_{0,j}(0) \tag{5.23}$$

And if the $\Delta C>0$, then the MUE can achieve larger network-wide capacity by accessing the FAP n^*. Proved; □

According to (5.17) and (5.21), we develop an access scheme for the MUEs, which is given in Algorithm 9. With this access scheme, each MUE chooses the BS (i.e., the MBS or an FAP) with the best channel condition to access, as given in Lines 2–3 in Algorithm 9. For the MUEs that falls within the coverage of each FAP but are connected to the MBS (as determined in Lines 2–3), we next examine if switching such MUEs to the corresponding FAP can achieve further gains in the overall network capacity, as in Line 7, and switch such MUEs to the corresponding FAP if this is the case, as in Lines 8–9. It can be verified that the complexity of Algorithm 9 is $\mathcal{O}(L_0N)$.

Once the cell associations for the MUEs are determined by Algorithm 9 (note that for the FUEs, the FAP associations are already determined; see (5.9)), we next develop a greedy algorithm for spectrum resource allocation for the users. The goal of this algorithm is to greedily maximize the overall capacity of the system under the QoS constraint (5.14). The algorithm is shown in Algorithm 15, where $G_{i,j}(k)$ and ψ_i are defined in (5.18) and (5.19), respectively; ϕ_i is the spectrum

Algorithm 5: Access Scheme

for $j = 0 \rightarrow L_0$ **do**
    $i = \arg\max_{0 \le i \le N} \{G_{i,j}(0)\}$
    $\rho_{i,j}(0) = 1$
for $j = 0 \rightarrow L_0$ **do**
    if $\rho_{0,j}(0) == 1$ **then**
        $n^* = \arg\max_{1 \le i \le N} \{G_{i,j}(0)\}$
        if $\sum_{i \in N \setminus n^*} \frac{\psi_i}{G_{0,j}(0)} - \frac{\psi_{n^*}}{G_{n^*,j}(0)} + \frac{\psi_{n^*}}{G_{0,j}(0)} > 0$ **then**
            $\rho_{n^*,j}(0) = 1$
            $\rho_{0,j}(0) = 0$

needed by FAP i to satisfy the QoS requirements of all the UEs it serves; $B(\psi_i)$ is the spectrum resource of the UE corresponding to ψ_i. The algorithm first determines the bandwidth needed for satisfying the QoS requirement for each UE, and then allocates the spectrum to each BS according to the number of UEs it serves, which is given by Algorithm 9.

The spectrum B is allocated as follows. If $\psi_0 \geq \sum_{i=1}^{N} \psi_i$, allocate the extra spectrum to the MBS and the MBS then allocates it to the MUE connecting to it and having the best channel condition. In this case, as the spectrum resource allocated to the FAPs is determined by the FAP that needs the most spectrum resource to meet the QoS requirements of the UEs connecting to it, some other FAPs may still have some extra spectrum for allocation and they allocate the extra spectrum to the UEs with the best channel condition among those that connect to it. On the other hand, if $\psi_0 < \sum_{i=1}^{N} \psi_i$, the extra spectrum is allocated to the FAPs, and the FAPs will allocate the extra spectrum to the UEs with best channel condition among those connecting to it. It can be verified that the complexity of Algorithm 15 is also $\mathcal{O}(L_0 N)$.

5.4.2 Performance Upper Bound

We next derive a performance upper bound for the overall network capacity. According to (5.15), we can derive the upper bound as in (5.24).

Algorithm 6: Spectrum Allocation

for $j = 0 \rightarrow L_0$ **do**
  if $\rho_{0,j}(0) == 1$ **then**
    $B_{0,j}(0) = \mathcal{C}/G_{0,j}(0)$
for $i = 1 \rightarrow N$ **do**
  for $j = 0 \rightarrow L_i$ **do**
    $B_{i,j}(i) = \mathcal{C}/G_{i,j}(i)$
  for $j = 0 \rightarrow L_0$ **do**
    if $\rho_{i,j}(0) == 1$ **then**
      $B_{i,j}(0) = \mathcal{C}/G_{i,j}(0)$
if $\psi_0 \geq \sum_{i=1}^{N} \psi_i$ **then**
  $B(\psi_0) = B(\psi_0) + B - \phi_0 - \max_{\{1 \leq i \leq N\}} \phi_i$ **for** $i = 1 \rightarrow N$ **do**
    $B(\psi_i) = B(\psi_i) + \max_{\{1 \leq i \leq N\}} \phi_i - \phi_i$
else
  for $i = 1 \rightarrow N$ **do**
    $B(\psi_i) = B(\psi_i) + B - \phi_0 - \phi_i$

$$
\begin{aligned}
& \sum_{i=0}^{N} C_i \\
= \quad & \sum_{j \in \mathcal{L}_0} \rho_{0,j}(0) B_{0,j}(0) \log_2(1 + \varepsilon_{0,j}(0)) + \\
& \sum_{i=1}^{N} \left\{ \sum_{j \in \mathcal{L}_0} \rho_{i,j}(0) B_{i,j}(0) \log_2(1 + \varepsilon_{i,j}(0)) + \sum_{j \in \mathcal{L}_i} \rho_{i,j}(i) B_{i,j}(i) \log_2(1 + \varepsilon_{i,j}(i)) \right\} \\
& \leq \sum_{j \in \mathcal{L}_0} \rho_{0,j}(0) B_{0,j}(0) \psi_0 + \sum_{i=1}^{N} \left\{ \sum_{j \in \mathcal{L}_0} \rho_{i,j}(0) B_{i,j}(0) \psi_i + \sum_{j \in \mathcal{L}_i} \rho_{i,j}(i) B_{i,j}(i) \psi_i \right\} \\
= \quad & \psi_0 (B - \phi') + \phi' \sum_{i=1}^{N} \psi_i \\
= \quad & \psi_0 B + \phi' \left(\sum_{i=1}^{N} \psi_i - \psi_0 \right) \\
\leq \quad & B\psi_0 + B \max \left\{ \sum_{i=1}^{N} \psi_i - \psi_0, 0 \right\} = B \max \left\{ \sum_{i=1}^{N} \psi_i, \psi_0 \right\}. \qquad (5.24)
\end{aligned}
$$

In (5.24), first inequality is due to the definition of ψ_i, i.e., as the maximum $G_{i,j}(k)$. The second inequality is due to the fact that $(B - B_0) \leq B$ (i.e., $B_0 \geq 0$). This result is summarized in the following theorem.

Theorem 5.2. *The network-wide capacity achieved by the proposed algorithms is upper bounded as follows.*

$$\sum_{i=0}^{N} C_i \quad \leq \quad B \max \left\{ \sum_{i=1}^{N} \psi_i, \psi_0 \right\} \tag{5.25}$$

5.5 scenario with overlapped FAPs

We have discussed the scenario that all FAPs are not overlapped. However, in practical scenario, the deployment of FAPs can not be well organized and avoid overlapping, as FAPs are deployed by users. In this case, the coverage of some FAPs are overlapped in the scenario with overlapped FAPs. With overlapped FAPs, not all FAPs are able to reuse spectrum simutaneously because of the interference among them. This makes the access scheme and the spectrum allocation much more complicated.

In the introduction, we have mentioned that when the coverage of FAPs are overlapped, we can form the overlapped FAPs into FAP clusters, and then allocate frequency spectrum for clusters. Here, we introduce the access scheme and the spectrum allocation for the scenario with overlapped FAPs.

5.5.1 Accesss Scheme In Scenario With Overlapped FAPs

In 5.17, we have shown that the frequency efficiency each MUE can be improved if it access an MBS or FAP that offers the best SINR. Besides, as shown in 5.21, sometimes the whole system will achieve better overall performance if some MUE choose to access a FAP even if the MBS offers greater SINR. However, as evaluating the overall performance of scenario with overlapped

FAPs is much more complex that of non-overlapped scenario, we simplify the access scheme of MUEs to be accessing the MBS or FAP with stronger SINR at here.

5.5.2 Spectrum Allocation For Scenario With Overlapped FAPs

Although the interference deteriorates the spectrum reuse between overlapped FAPs, the frequency reuse is still feasible between FAPs not overlapped. Hence, if we divide the FAPs into clusters that there is no interference between clusters, then the same spectrum can be reused between clusters. A cluster is group of FAPs that there exists interference between FAPs in a cluster, and there is no interference between FAPs from different clusters. Here, we give a mathematical definition for clusters.

Lemma 5.1. *If two FAPs A and B can interfere each other, then they are communicate, denote at A↔B. And the communicate relationship is transitive. That is, If A↔B, and B↔C, then A↔C. All FAPs are divided into subsets $\mathcal{S}_1$, $\mathcal{S}_2$, ..., $\mathcal{S}_k$, called clusters , such that any two FAPs within the same cluster communicate, but FAPs from different clusters do not. Where $\mathcal{S}_1 \cup \mathcal{S}_2 \cup, ..., \cup \mathcal{S}_k = \{1, 2, .., N\}$, and $\mathcal{S}_1 \cap \mathcal{S}_2 \cap, ..., \cap \mathcal{S}_k = \varnothing$.*

Where for FAPs have no interference with other FAPs, each of them form a cluster with only one FAP.

As FAPs are divided into clusters and there is no interference between FAPs from different clusters, spectrum can be simultaneously reused by clusters. However, a FAPs is not necessarily interfering with all others FAPs in the same cluster.

For example, in figure.5.1, there are 4 FAPs. FAP_1 is interfering with FAP_2, and FAP_3 is interfering with FAP_2 and FAP_4. In this case, FAP_1 and FAP_2 can not reuse the same frequency spectrum simultaneously, and FAP_2, FAP_3 and FAP_4 can not reuse the same frequency spectrum simultaneously. However, if we divide them into two group, $\{FAP_1, FAP_3\}$ and $\{FAP_2, FAP_4\}$, then FAPs in the same group can use the same frequency spectrum simultaneously. Hence, it is necessary for us to find a feasible and effective algorithm to divide every cluster with more than

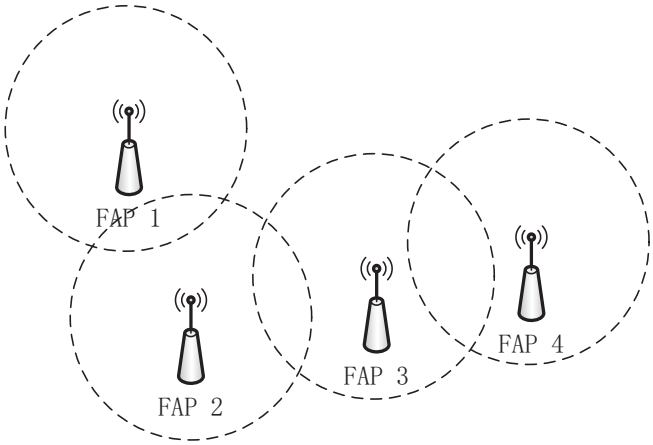

Figure 5.1: example of a cluster with 4 FAPs.

one FAPs into groups that all FAPs in the same group are able to use the same frequency spectrum simultaneously.

Here we define the index variable $V_{i,j}$ as

$$V_{i,j} = \begin{cases} 1, & \text{the coverage of } FAP_i \text{ and } FAP_j \text{ is overlapped} \\ 0, & \text{otherwise,} \end{cases} \quad \forall\, i,k \in \mathcal{N}. \tag{5.26}$$

According to the definition of $V_{i,j}$, the constraint of the spectrum allocation in a cluster can be described as $\sum_{j \in S_k} B_j(t) L_{i,j} \leq 1$, for $i \in S_k$, and $S_k \subset N$. Where $B_j(t)$ means certain frequency spectrum point is assigned to FAP_j at time t. Then the spectrum allocation problem can be formulated as follow:

$$\text{maximize} \quad C_0 + \sum_{n=0}^{k-1} C_{\mathcal{S}_n} \tag{5.27}$$

subject to:

$$constraint(5.8), (5.9), (5.10), (5.11), (5.12) and (5.14) \tag{5.28}$$

$$\sum_{j \in \mathcal{S}_n} B_j(t) V_{i,j} \leq 1, for\ i \in \mathcal{S}_n. \tag{5.29}$$

The idea of solving the problem can be summarized into the following three steps:

1. Divide the all FAPs into clusters.
2. Divide each clusters into groups, where FAPs in the same group can reuse the same spectrum in the same time.
3. Allocate spectrum resource for each FAP

5.5.3 Solution Algorithms

Firstly, we try to divide FAPs into clusters. we may consider the topology of FAPs as undirected graph G = (V,E). Where V =1,...,N, representing the FAP nodes. V contains all edges between vertices, and a edge between two vertices means the two corresponding FAP nodes are overlapped. According to the definition of the clusters in lemma5.1, a FAP node can reach to all FAPs nodes among the cluster. So we can identify all nodes in a cluster, if we search for all reachable vertices from one vertex. Here, we adopt the idea of breadth-first search [62] to identify FAPs in the same cluster and then divide all FAPs nodes into clusters. The algorithm is presented as the algorithm 7:

In the algorithm, the graph G=(V, E) uses a adjacency-list to represent the edges. For each Vertex u $\in$ V, a Adj list is created to record the Adj vertices. That is, Adj[u] includes all vertices that connected to vertex u. The algorithm also uses a First-In-First-Out (FIFO) queue Q. The function ENQUEUE(Q,s) pushes s into Q, and the function DEQUEUE(Q) pushes a node out. The algorithm first labels all nodes to be white and initialize the FIFO queue Q. Then it select the node first mode as the source node for searching. After the search, all nodes in the same cluster of the source node are identified and labeled black. Then the algorithm select the next node that is still white to be the source and do search again. This process is repeated until all nodes are divided into clusters and labeled black. The function SEARCH(G,s,i) is a modified Breath-first search algorithm, readers may check chapter 22 of [62] for more information.

Algorithm 7: divide into cluster

```
1  cluster(G):
2  for each vertex u∈ G.V do
3      u.color = WHITE
4  Q = ∅
5  i = 0, k = 0 while G.V[k] ≠ Nil do
6      s = G.V[k]
7      if s.color = = WHITE then
8          SEARCH (G, s, i)
9      k++
10 SEARCH (G, s, i):
11 ENQUEUE (Q, s)
12 while Q ≠ ∅ do
13     u = DEQUEUE (Q)
14     for each v=G.Adj[u] do
15         if v.color == WHITE
16         v.color = GRAY
17         ENQUEUE(Q,v)
18     u.color = BLACK
19     cluster [i].add (u)
20 i++
```

After grouping nodes into clusters, we then need to divide nodes in each cluster into groups. The algorithm should ensure that each FAP nodes are not overlapped (not connected in the topology) with any other nodes among the same group. The algorithm is shown as algorithm(8)

Where we use G=(V,E) to represent the topology of FAPs in a cluster, as in algorithm(7). In the initialization, we label all vertices to WHITE. The algorithm select the first WHITE vertex s, then labels all adjacent vertices to BLACK. By repeating that, the algorithm select a group of vertices that not connected with each other, and the remaining vertices are BLACK, which means they are connected to at least one of the vertices picked up in the group. After a group of vertices are selected, we deleted them from the gragh, and label the remaining vertices WHITE again. The process is repeated until all vertices are selected. When the algorithm is finished, all vertices are divided into groups, in which each vertex is not connected with other vertex in the same group.

Then we proposed an algorithm to allocate the frequency spectrum. Similar to the non-overlapped scenario, the objective of the algorithm is to maximize the network-wide capacity

Algorithm 8: divide into groups

```
1  grouping (G)
2  k=0 while G.V[k] ≠ 0 do
3      s = G.V[k]
4      s.color = WHITE
5  i=0
6  while G.V ≠ ∅ do
7      k=0
8      while G.V[k] ≠ Nil do
9          s=G.V[k]
10         if s.color == WHITE then
11             for each v = G.Adj[s] do
12                 V.color = BLACK
13             Group[i].add(s)
14             delete(s)
15     k = 0
16     while G.V[k] ≠ Nil do
17         s=G.V[k]
18         s.color = WHITE
```

and the guarantee the QoS requirement of each UE. For the scenario with overlapped FAPs, the frequency allocation is much more complex that in scenario with no overlapping FAPs, as the spectrum allocation scheme is not independent among FAPs in a cluster. Firstly, we analysis the spectrum allocation in one cluster. Assume that in cluster $\mathcal{S}_n$, there are m FAPs, and they can be divided into k groups $Group_0, Group_1, ...Group_{k-1}$. Assume that frequency spectrum B' is allocated to the cluster and the QoS requirement is fulfilled. Then if extra band frequency spectrum ΔB can be allocated to cluster $\mathcal{S}_n$, the added throughput ΔC can be represented as:

$$\Delta C = \sum_{i \in S_n} \Delta C_i \tag{5.30}$$

$$= \sum_{j=0}^{j=k-1} \sum_{i \in Group_j} \Delta C_i \tag{5.31}$$

$$= \sum_{j=0}^{j=k-1} \sum_{i \in Group_j} \Delta B_j \psi_i \tag{5.32}$$

$$\leq \Delta B \max(\sum_{i \in Group_j} \psi_i) \tag{5.33}$$

Where ΔB_j is the extra spectrum allocated to $Group_j$, and $\sum_{j=0}^{j=k-1} \Delta B_j = \Delta B$. Besides, ψ_i is defined in equation(5.19).

So in each cluster, the extra spectrum resource should be allocated to the group with maximum $\sum_{i \in Group_j} \psi_i$, to get better network-wide capacity.

For the overall throughput, if there are extra spectrum ΔB, then the added throughput can be represented as:

$$\Delta C \tag{5.34}$$
$$= \Delta C_0 + \sum_{n=0}^{k-1} \Delta C_{Sn} \tag{5.35}$$
$$\leq (\Delta B_0)\psi_0 \tag{5.36}$$
$$+ \sum_{n=0}^{k-1} (\Delta B - \Delta B_0) \max(\sum_{i \in Group_j, Group_j \subset S_n} \psi_i) \tag{5.37}$$
$$\leq \Delta B \max\{\psi_0, \sum_{n=0}^{k-1} \max(\sum_{i \in Group_j, Group_j \subset S_n} \psi_i)\} \tag{5.38}$$

We can notice that if

$$\psi_0 > \sum_{n=0}^{k-1} \max(\sum_{i \in Group_j, Group_j \subset S_n} \psi_i) \tag{5.39}$$

holds, the we can get better network-wide capacity if we allocate the extra spectrum resource to the MBS. Otherwise, we can get better network-wide capacity if we allocate the extra spectrum resource to the FAPs.

Hence, the spectrum allocation algorithm can be represented as algorithm(9)

In the algorithm(9), we assume that we have already applied the algorith (7) and (8). That is, we have already divided the FAPs into clusters and groups. The basic idea of the algorithm is that

Algorithm 9: Spectrum allocation for the overlapped scenario

$\phi_{max} = 0$
for *every cluster* S_n **do**
  $\phi(S_n) = 0$
  for *every group m* $\in$ *cluster* S_n **do**
    $\phi(S_n, m) = 0$
    for *Every FAP i in group m* **do**
      if $\phi_i > \phi(S_n, m)$ **then**
        $\phi(S_n, m) = \phi_i$
      for $j = 1 \rightarrow \mathcal{L}_0$ **do**
        if $\rho_{i,j}(0) == 1$ **then**
          $B_{i,j}(0) = \mathcal{C}/G_{i,j}(0)$
      for $j = 1 \rightarrow \mathcal{L}_i$ **do**
        $B_{i,j}(i) = \mathcal{C}/G_{i,j}(i)$
    $\phi(S_n) = \phi(S_n) + \phi(S_n, m)$
    for *every FAP i in group m* **do**
      if $\phi_i < \phi(S_n, m)$ **then**
        $B(\psi_i) = B(\psi_i) + \psi(S_n, m) - \psi_i$
  if $\phi(S_n) > \phi_{max}$ **then**
    $\phi_{max} = \phi(S_n)$
for $j = 1 \rightarrow \mathcal{L}_0$ **do**
  if $\rho_{0,j}(0) == 1$ **then**
    $B_{0,j}(0) = \mathcal{C}/G_{0,j}(0)$
if $\psi_0 > \sum_{n=0}^{k-1} \max(\sum_{i \in Group_j, Group_j \subset S_n} \psi_i)$ **then**
  $B(\psi_0) = B(\psi_0) + B - \phi_0 - \phi_{max}$
  for *every cluster* S_n **do**
    if $\phi(S_n) < \phi_{max}$ **then**
      $m = \arg\max_j \{\psi_i | i \in Group_j, Group_j \subset S_n\}$
      for *every FAP i in group m* **do**
        $B(\psi_i) = B(\psi_i) + \phi_{max} - \phi(S_n)$
else
  for *every cluster* S_n **do**
    $m = \arg\max_j \{\psi_i | i \in Group_j, Group_j \subset S_n\}$ **for** *every FAP i in group m* **do**
      $B(\psi_i) = B(\psi_i) + B - \phi_0 - \phi(S_n)$

we first need to ensure the QoS requirement of each UE. Then we should ensure that different clusters reuse the same spectrum resource, and in each clusterm FAPs in the same group should share

the same spectrum and the FAPs from different groups must avoid occupying the same spectrum to avoid interference. Lastly, we also need to follow the idea discussed in (5.30) and (5.34).

In algorithm(9), from line(7) to line (8), we calculate the minimum spectrum resource needed to ensure the QoS requirement of all FAPs in each group. From line(9) to line(13), we allocate spectrum for UEs that access to FAPs to meet their QoS requirement. Line(14) calculate the minimum spectrum resource needed to ensure the QoS requirement for each cluster. From line(15) to line(17) we allocate some extra spectrum to the UEs with the greatest ψ_i in each group to make FAPs in the same group share the same spectrum resource. Line(18) to line(19) calculate the minimum spectrum resource needed to ensure the QoS requirement for all clusters. Line(20) to line(22) allocates the minimum spectrum resource for MUEs that access the MBS to ensure the QoS requirement. In line(23), we check the inequation (5.39). If inequation(5.39) holds, we allocate the extra spectrum to the MUE with best SINR (greatest ψ_0) among MUEs that access the MBS in line(24). And from line(25) to line(29), we allocate some spectrum resource to some UEs access to FAPs with THE best SINR to the make all clusters share the same spectrum resource. Otherwise, we allocate the extra spectrum resource to some UEs access to FAPs with best SINR and make them share the same spectrum resource(between line(31) and line(33).

5.5.4 Performance Upper Bound

Based on the discuss in (5.30) and (5.34), the upper bound of the network-wide capacity can be derived as follow:

$$C = C_0 + \sum_{n=0}^{k-1} C_{Sn} \tag{5.40}$$

$$\leq B_0\psi_0 + \sum_{n=0}^{k-1}(B - B_0)\max(\sum_{i\in Group_j, Group_j\subset S_n}\psi_i) \tag{5.41}$$

$$\leq B\max\{\psi_0, \sum_{n=0}^{k-1}\max(\sum_{i\in Group_j, Group_j\subset S_n}\psi_i)\} \tag{5.42}$$

5.6 Performance Evaluation

5.6.1 Scenario With Non-overlap FAP

We evaluate the performance of the proposed scheme with MATLAB simulations. Specifically, we compared the proposed algorithms with the access scheme and resource allocation mechanism (termed OA scheme) presented in [53], as well as the OA scheme enhanced with our proposed resource allocation algorithm (OA-PRA). In OA, the MUEs decide to access the MBS or an FAP that provides the best SINR; if an MUE chooses to access an FAP, the FAP will be allocated with the corresponding spectrum resource. In the following simulations, the network has a total spectrum resource of $B = 20$ MHz. The coverage of the MBS is 500 m and the coverage of the FAPs are 50 m. In addition, we assume each FAP has one FUE and there are a large number of MUEs. The channel models are defined in (5.2) and (5.3), respectively.

In Fig. 5.2, we evaluate the impact of the number of FAPs on the total capacity of the system. In the simulation, there are 100 MUEs, the QoS requirement $\mathcal{C}$ is set to 400 Kbps. As shown in the figure, the total capacity increases as more FAPs are deployed. For OA, the total capacity increases slightly with the number of FAPs N. In the proposed algorithm and OA-PRA, the total capacity increases greatly with N. This is because that more resources are allocated to users with better SINR, and resources can be spatially reused among the FAPs. The proposed algorithm achieves better performance than OA-PRA when there are more than one FAPs. After all, the proposed access scheme has taken into account spatial reuse among FAPs. For the one FAP scenario, OA-PRA and the proposed algorithms achieve an equal total capacity. Actually OA-PRA is equivalent to the proposed algorithm when there is only one FAP in the system. In short, the proposed algorithm achieves considerable network capacity gains than OA, due to the integration of access control and resource allocation. We also find that the upper bound given in Theorem 5.2 is quite tight for the range of FAP numbers examined in this study.

In Fig. 5.3, we evaluate the impact of the QoS requirement $\mathcal{C}$ on the total capacity of the system. In this simulation, there are 100 MUEs and 4 FAPs. From the figure, we notice that when

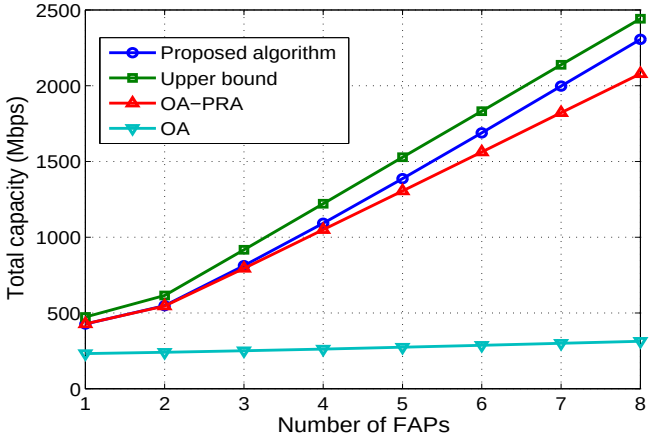

Figure 5.2: Number of FAPs versus total capacity.

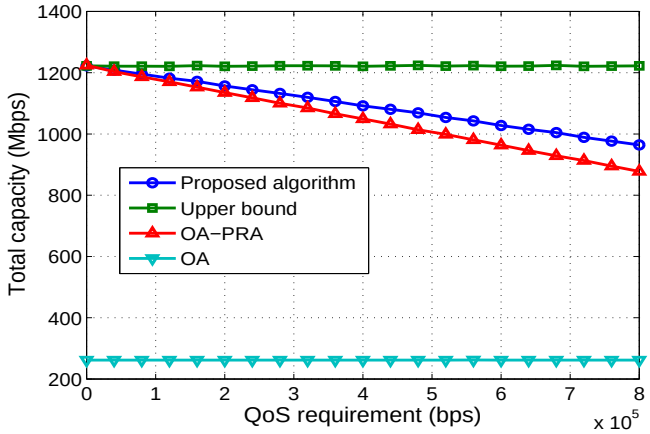

Figure 5.3: QoS requirement versus total capacity.

QoS requirement is 0, the upper bound, proposed algorithm and OA-PRA achieve the same capacity. Actually, when there is no QoS requirement, in the proposed scheme and OA-PRA, the system allocates all the spectrum resource to the UEs that bring larger capacity gains, hence achieving the upper bound given in Theorem 5.2. With increased QoS requirement, the performance of the proposed scheme and OA-PRA degrades, but is still much higher than that of OA. This is because that a more stringent QoS requirement forces the system to allocate more spectrum resource to UEs with a lower SINR to ensure that their QoS requirements are met. Hence, there is a balance between fairness and efficiency, as can be seen from this study. The proposed scheme always achieves better performance than that of OA-PRA and OA, and the gain gets larger when the QoS requirement is increased.

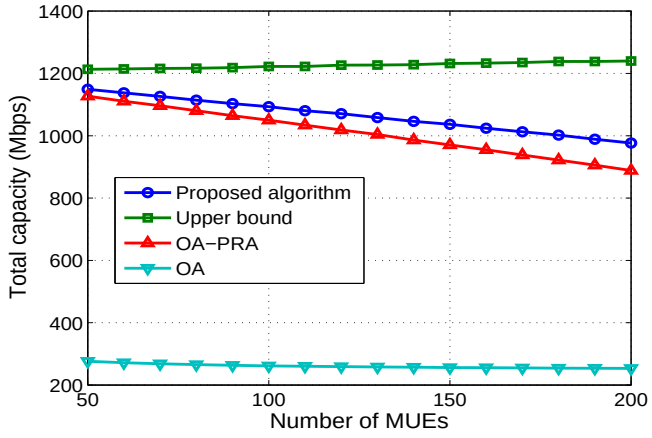

Figure 5.4: Number of MUEs versus total capacity.

In Fig. 5.4, we examine the impact of the number of UEs on the total capacity of the system. In this simulation, there are 4 FAPs, and the QoS requirement $\mathcal{C}$ is set to 400 Kbps. It can be seen that the proposed scheme always outperforms both OA and OA-PRA. In addition, the performance of the proposed scheme and OA-PRA get worse with more MUEs are enabled. The reason is similar to that in Fig. 5.3. With more MUEs, the system needs to allocate more spectrum resource to the UEs with lower SINRs and hence less spectrum resource will be available for the MUEs with good channels.

5.6.2 Scenario With Overlapped FAPs

We also evaluate the performance of proposed scheme for the the scenario with overlapped FAPs. We compared the performance with the proposed algorithm on non-overlap case. We also compared the proposed algorithm with OA scheme [53]. As the OA-PRA is may not be applied to the overlapped FAP case, it is not included in the comparation. In the simulation, there are 6 FAPs, the topology can be summerized as: FAP_1 is not overlapped as any FAPs; FAP_2 is overlapping with FAP_3, and FAP_3 is overlapping with FAP_4; FAP_5 is overlapping with FAP_6. Other settings are the same with the non-overlap case, if not specifically pointed out. In figure. 5.5, we examine the impact of the number of UEs on total capacity of the system. In this simulation, the QoS requirement $\mathcal{C}$ is set to 400Kbps. It can be seen that the performance of proposed algorithm

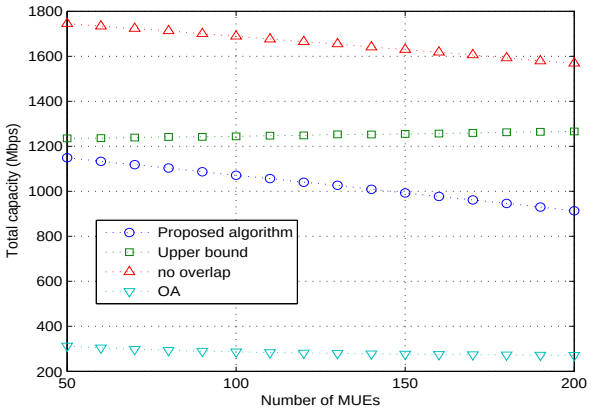

Figure 5.5: Number of MUEs versus total capacity.

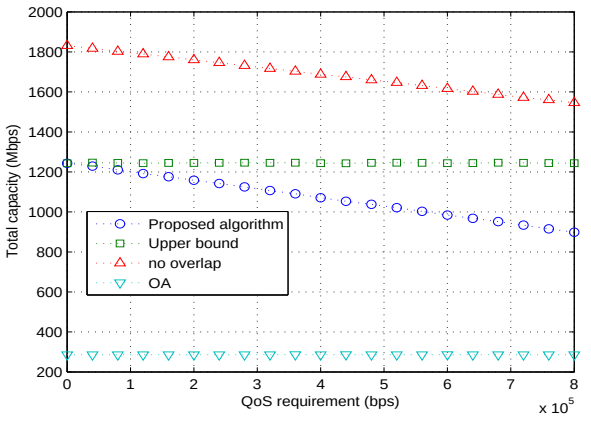

Figure 5.6: QoS requirement versus total capacity.

on overlapping case is worse that that on non-overlapping case, but it still outperforms the OA scheme. And it is simular to the simulation in Figure. (5.6), where we examine the impact of different QoS requirement on the system performance.

5.7 Conclusion

In this work, we studied the access strategy of MUEs and spectrum resource allocation for the FAPs in a two-tier femtocell network. We considered the dedicated channel and open access deployment scenario, and used spectrum resource as incentives to encourage FAPs to serve more MUEs. The objective is to maximize the overall performance of the network while guaranteeing

the QoS requirement for the users. To solve the formulated MINLP problem, we proposed an algorithm to decide the access policy for the MUEs, and an algorithm for allocation of spectrum resources to the FAPs. An upper bound was derived for the total capacity achieved by the proposed algorithms. The bound and proposed algorithms were evaluated with simulations and shown to outperform an existing scheme.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

In this work, we aim to enhance the computational capacity and wireless data transmitting rate, and stabilize of computing and transmission queues of mobile devices. Based on lyapunove optimization, we balance the energy consumption and the queue length in dynamic cloud offloading scheme. We also studied the distributed online auction for sharing unlicensed bands among LTE-unlicensed BS's, scheduling algorithms in full-duplex enabled multi-channel WLAN and access strategy of MUEs and spectrum resource allocation for the FAPs in a two-tier femtocell network, in order to improve the connectivity of mobile devices while minimizing the energy consumption.

In chapter 2, we proposed a scheduling scheme for energy-efficient cloud offloading for muti-core mobile devices, while considering downloading the cloud execution output in the model. studied the energy delay trade-off in cloud offloading for multi-core mobile devices. Based on Lyapunov optimization, we developed an online algorithm that does not require information about stationary distribution of applications and the network condition, making it amenable to real-time implementation for practical scenarios. We proved theoretical bounds for the proposed algorithm and validated its performance with trace-driven simulations.

In chapter 3, we studied distributed online auction for sharing unlicensed bands among LTE-unlicensed BS's to maximize the social welfare in each auction, while achieving the dual goal of minimizing the expected packet dropping rate and guarantee a maximum delay. Specifically, we propose Lyapunov optimization based schemes to evaluate the true value of unlicensed spectrum, to allocate RBs on unlicensed bands, and to decide when to drop packets based on current channel

condition, queue lengths, and delay of packets. We also proposed a truthful auction mechanism to integrate the schemes, which can maximize the overall social welfare and guarantee bounded drop rate and delay. The superior performance of the proposed algorithms over two benchmark schemes was validated with simulations.

In chapter 4, we proposed an online scheduling algorithm to jointly decide the channel assignment, transmission scheduling, half- or full-duplex transmission mode selection for each UE in a multi-channel ful-duplex WLAN. The proposed scheme was also based on Lyapunov optimization. We also proved the optimality of the proposed algorithm and derived upper bounds for the average queue length and energy consumption under the proposed algorithm. We evaluated the performance of the proposed algorithm with simulations. We showed that under the proposed algorithm, there was a trade-off between the average queue length and energy consumption under different V values.

In chapter 5, we studied the access strategy of MUEs and spectrum resource allocation for the FAPs in a two-tier femtocell network. We considered the dedicated channel and open access deployment scenario, and used spectrum resource as incentives to encourage FAPs to serve more MUEs. The objective is to maximize the overall performance of the network while guaranteeing the QoS requirement for the users. To solve the formulated MINLP problem, we proposed an algorithm to decide the access policy for the MUEs, and an algorithm for allocation of spectrum resources to the FAPs. An upper bound was derived for the total capacity achieved by the proposed algorithms. The bound and proposed algorithms were evaluated with simulations and shown to outperform an existing scheme.

6.2 Future Work

With the unprecedented growth in wireless data, wireless operators are in critical need of more spectrum for higher capacity. To meet the so-called 1000x mobile data challenge [87], extending LTE to the unlicensed spectrum, as specified in LTE Rel-10 – Rel-13 [83, 84], has recently gained significant attention [83, 87, 88, 90]. However, there are two main challenges to the success of the

so-called *LTE-unlicensed* technology. First, the unlicensed bands are already occupied by many existing wireless networks (e.g., WiFi). It is essential to enable the coexistence of LTE-unlicensed with existing unlicensed band users, i.e., to avoid significant performance degradation to existing users while achieving high capacity gains with LTE-unlicensed. Second, the interference in unlicensed bands is unpredictable, which is detrimental to the performance of LTE-unlicensed users. Hence, it is important to effectively manage the interference between LTE-unlicensed and existing users, and that among LTE-unlicensed users themselves.

To study the coexistence of LTE-unlicensed with existing unlicensed band users, some system level simulation studies have been reported in several recent works [88, 93, 94]. The simulation results show that the WiFi performance could be significantly degraded, while the LTE performance is only slightly affected. This is because WiFi uses Carrier Sensing Multiple Access (CSMA) to compete for channel access, while LTE adopts a centralized channel access control mechanism. WiFi usually keeps silent when sensing a busy channel continuously used by LTE. To protect existing unlicensed band users, requirements for clear channel assessment (CCA) and Listen Before Talk (LBT) are specified by European standardization bodies [95]. In LBT, a user equipment (UE) must perform CCA on the operating channel(s) before starting a transmission. The observing duration should be at least 20 μs.

Although the LTE performance may be only slightly affected by WiFi in some coexistence scenarios [93, 94], there could still be significant throughput degradations due to the inter-operator interference, when multiple LTE-unlicensed base stations (BS) of different operators are deployed in the same area [83]. There are two solutions to this problem: (i) make an agreement for the operators to allocate the unlicensed spectrum; or (ii) enable opportunistic access to unlicensed channels. The first solution may not be practical in most countries due to competition among operators and the lack of regulation for unlicensed bands [83], while the second solution is promising for effective unlicensed spectrum sharing.

For future works, we will study the following problems,

1. Investigating reasonable metrics of defining whether existing unlicensed band users is interfered by LTE-unlicensed.

2. Verifying whether the LBT can efficiently protect the existing unlicensed band users.

3. Investigating the capacity region of LTE-unlicensed users when coexist with existing unlicensed band users.

4. Studying spectrum sharing among LTU-unlicense BS's to avoid interference among LTE-unlicensed users.

Bibliography

[1] http://www.statista.com/statistics/201182/forecast-of-smartphone-users-in-the-us/

[2] http://www.statista.com/statistics/263795/number-of-available-apps-in-the-apple-app-store/

[3] Z. Jiang and S. Mao, "Energy delay trade-off in cloud offloading for multi-core mobile devices," in *Proc. IEEE GLOBECOM 2015*, San Diego, CA, Dec. 2015, pp.1–6.

[4] Y. Xu and S. Mao, "A survey of mobile cloud computing for rich media applications," *IEEE Wireless Commun.*, vol. 20, no. 3, pp. 46–53, June 2013.

[5] Y. Xu and S. Mao, "Mobile cloud media: State of the art and outlook," in *Mobile Computing over Cloud: Technologies, Services, and Applications*, Chapter 2, pp.18–38, J. Rodrigues, K. Lin, and J. Lloret (Editors), IGI Global: Hershey, PA, 2013.

[6] K. Kumar, J. Liu, Y.-H. Lu, and B. Bhargava, "A Survey of Computation Offloading for Mobile Systems," *Mobile Networks and Applications*, vol. 18, no. 1, pp.129–140, Feb. 2013.

[7] S. Kosta and et al., "ThinkAir: Dynamic resource allocation and parallel execution in the cloud for mobile code offloading," in *Proc. IEEE INFOCOM 2012*, Orlando, FL, Mar. 2012, pp. 945–953.

[8] M. Chen, Y. Zhang, Y. Li, S. Mao, and V.C.M. Leung, "EMC: Emotion-aware mobile cloud computing in 5G," *IEEE Network*, vol.29, no.2, pp.32–38, Mar./Apr. 2015.

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

[10] T. D. Burd and R. W. Brodersen, "Processor design for portable systems," *Kluwer J. VLSI Signal Process. Syst.*, vol. 13, no. 2/3, pp. 203–221, Aug. 1996.

[11] M. Barbera, S. Kosta, A. Mei, and J. Stefa, "To offload or not to offload? The bandwidth and energy costs of mobile cloud computing," in *Proc. IEEE INFOCOM 2013*, Turin, Italy, Apr. 2013, pp. 1285–1293.

[12] A. J. Nicholson and et al., "Improved access point selection," in *Proc. ACM MobiSys'06*, Uppsala, Sweden, June 2006, pp. 233–245.

[13] Y. Wang, S. Mao, and R. M. Nelms, "An online algorithm for optimal real-time energy distribution in smart grid," *IEEE Trans. Emerg. Topics Comput.*, vol. 1, no. 1, pp. 10–21, July 2013.

[14] Z. Jiang and S. Mao, “Online channel assignment, transmission scheduling, and transmission mode selection in multi-channel full-duplex wireless LANs,” in *Proc. WASA 2015*, Qufu, China, Aug. 2015, pp. 1–10.

[15] P. Shu and et al., “eTime: Energy-efficient transmission between cloud and mobile devices,” in *Proc. IEEE INFOCOM 2013*, Turin, Italy, Apr. 2013, pp. 195–199.

[16] H. W. Kuhn, “The hungarian method for the assignment problem,” *Naval Research Logistics Quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955. [Online]. Available: `http://dx.doi.org/10.1002/nav.3800020109`

[17] J. Huang and et al., “A close examination of performance and power characteristics of 4G LTE networks,” in *Proc. ACM MobiSys’12*, Low Wood Bay, UK, June 2012, pp. 225–238.

[18] W. Zhang, Y. Wen, Z. Chen, and A. Khisti, “QoE-driven cache management for HTTP adaptive bit rate streaming over wireless networks,” *IEEE Trans. Multimedia*, vol. 15, no. 6, pp. 1431–1445, Oct. 2013.

[19] W. Zhang and et al., “Energy-optimal mobile cloud computing under stochastic wireless channel,” *IEEE Trans. Wireless Commun.*, vol. 12, no. 9, pp. 4569–4581, Sept. 2013.

[20] Y. Jin, Y. Wen, H. Hu, and M.-J. Montpetit, “Reducing operational costs in cloud social TV: An opportunity for cloud cloning,” *IEEE Trans. Multimedia*, vol. 16, no. 6, pp. 1739–1751, Oct. 2014.

[21] B.-G. Chun and et al., “CloneCloud: Elastic execution between mobile device and cloud,” in *Proc. EuroSys’11*, Salzburg, Austria, Apr. 2011, pp. 301–314.

[22] D. Huang, P. Wang, and D. Niyato, “A dynamic offloading algorithm for mobile computing,” *IEEE Trans. Wireless Commun.*, vol. 11, no. 6, pp. 1991–1995, June 2012.

[23] Y. Wen, W. Zhang, and H. Luo, “Energy-optimal mobile application execution: Taming resource-poor mobile devices with cloud clones,” in *Proc. IEEE INFOCOM 2012*, Orlando, FL, Mar. 2012, pp. 2716–2720.

[24] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, “The case for VM-based cloudlets in mobile computing,” *IEEE Pervasive Computing*, vol.8, no.4, pp.14–23, Oct./Dec. 2009.

[25] E. Cuervo, A. Balasubramanian, D.-K. Cho, A. Wolman, S. Saroiu, R. Chandra, and P. Bahl, “MAUI: Making smartphones last longer with code offload,” in *Proc. ACM MobiSys’10*, San Francisco, CA, Mar. 2010, pp.49–62.

[26] B.-G. Chun, S. Ihm, P. Maniatis, M. Naik, and A. Patti, “CloneCloud: Elastic execution between mobile device and cloud,” In *Proc. ACM EuroSys’11*, Salzburg, Austria, Apr. 2011, pp.301–314.

[27] A. Leff, and J.T. Rayfield, “Integrator: An architecture for an integrated cloud/on-premise data-service,” in *Proc. IEEE ICWS’15*, New York, NY, June/July 2015, pp.98–104.

[28] X. Wang, J. Wang, X. Wang, and X. Chen, "Energy and delay tradeoff for application offloading in mobile cloud computing," in *IEEE Systems J.*, vol.PP, no.99, pp.1–10, Aug. 2015.

[29] C. Xu, Y. Qiao, B. Lee, and N. Murray, "Energy consumption of mobile offloading for JavaScript applications," in *2015 Irish Signals and Systems Conference*, Carlow, Ireland, June 2015, pp.1–6.

[30] Y.-S. Chen, C.-S. Hsu, T.-Y. Juang, and H.-H. Lin, "An energy-aware data offloading scheme in cloud radio access networks," in *Proc. IEEE WCNC'15*, New Orleans, LA, Mar. 2015, pp.1984–1989.

[31] Z. Chang, J. Gong, Z. Zhou, T. Ristaniemi, and Z. Niu, "Resource allocation and data offloading for energy efficiency in wireless power transfer enabled collaborative mobile clouds," in *Proc. IEEE INFOCOM'15 WKSHPS*, Hong Kong, China, Apr./May 2015, pp.336–341.

[32] W. Enck, P. Gilbert, S. Han, V. Tendulkar, B.-G. Chun, L.P. Cox, J. Jung, P. McDaniel, and A.N. Sheth, "TaintDroid: An information-flow tracking system for realtime privacy monitoring on smartphones," *Proc. USENIX OSDI'10*, Vancouver, BC, Canada, Oct. 2010, pp.1–6.

[33] A.Y. Ding, B. Han, Y. Xiao, P. Hui, A. Srinivasan, M. Kojo, and S. Tarkoma, "Enabling energy-aware collaborative mobile data offloading for smartphones," in *Proc. IEEE SECON'13*, New Orleans, LA, June 2013, pp.487–495.

[34] K. Kumar and Y.-H. Lu, "Cloud computing for mobile users: Can offloading computation save energy?" *IEEE Computer*, vol.43, no.4, pp.51–56, Apr. 2010.

[35] M. Altamimi, A. Abdrabou, K. Naik, and A. Nayak, "Energy cost models of smartphones for task offloading to the cloud," *IEEE Trans. Emerging Topics Computing*, vol.3, no.3, pp.384–398, Sept. 2015.

[36] J. I. Choi, M. Jain, K. Srinivasan, P. Levis, and S. Katti, "Achieving single channel, full duplex wireless communication," in *Proc. ACM MobiCom'10*, Chicago, IL, Sept. 2010, pp. 1–12.

[37] D. Bharadia, E. McMilin, and S. Katti, "Full duplex radios," *SIGCOMM Comput. Commun. Rev.*, vol. 43, no. 4, pp. 375–386, Aug. 2013.

[38] S. Gollakota and D. Katabi, "Zigzag decoding: Combating hidden terminals in wireless networks," in *Proc. ACM SIGCOMM'08*, Seattle, WA, Aug. 2008, pp. 159–170.

[39] M. Jain, J. I. Choi, T. Kim, D. Bharadia, S. Seth, K. Srinivasan, P. Levis, S. Katti, and P. Sinha, "Practical, real-time, full duplex wireless," in *Proc. ACM MobiCom'11*, Las Vegas, NV, Sept. 2011, pp. 301–312.

[40] M. Feng, S. Mao, and T. Jiang, "Joint duplex mode selection, channel allocation, and power control for full-duplex cognitive femtocell networks," *Elsevier Digital Commun. Netw. J.*, vol. 1, no. 1, pp. 30–44, Feb. 2015.

[41] Y. Wang and S. Mao, “Distributed power control in full duplex wireless networks,” in *Proc. IEEE WCNC’15*, New Orleans, LA, Mar. 2015, pp. 1–6.

[42] S. Goyal, P. Liu, S. Panwar, R. DiFazio, R. Yang, J. Li, and E. Bala, “Improving small cell capacity with common-carrier full duplex radios,” in *Proc. IEEE ICC’14*, Sydney, Australia, June 2014, pp. 4987–4993.

[43] X. Xie and X. Zhang, “Does full-duplex double the capacity of wireless networks?” in *Proc. IEEE INFOCOM’14*, Toronto, Canada, Apr. 2014, pp. 253–261.

[44] M. Neely, E. Modiano, and C. Rohrs, “Dynamic power allocation and routing for time-varying wireless networks,” *IEEE J. Sel. Areas Commun.*, vol. 23, no. 1, pp. 89–103, Jan. 2005.

[45] K. Kar, X. Luo, and S. Sarkar, “Throughput-optimal scheduling in multichannel access point networks under infrequent channel measurements,” in *Proc. IEEE INFOCOM’07*, Anchorage, AK, May 2007, pp. 1640–1648.

[46] Y. Huang, S. Mao, and R. M. Nelms, “Adaptive electricity scheduling in microgrids,” *IEEE Trans. Smart Grid*, vol. 5, no. 1, pp. 270–281, Jan. 2014.

[47] H. W. Kuhn, “The Hungarian method for the assignment problem,” *Naval Research Logistics Quarterly*, vol. 2, no. 1/2, pp. 83–97, Mar. 1955.

[48] Michael J. Neely, “Stability and Probability 1 Convergence for Queueing Networks via Lyapunov Optimization,” Journal of Applied Mathematics, vol. 2012, Article ID 831909, 35 pages, 2012. doi:10.1155/2012/831909

[49] A. Adhikary, V. Ntranos and G. Caire, ”Cognitive femtocells: Breaking the spatial reuse barrier of cellular systems,” *Information Theory and Applications Workshop (ITA)*, 2011, La Jolla, CA, 2011, pp. 1-10.

[50]

[51] V. Chandrasekhar and J. G. Andrews, ”Spectrum allocation in tiered cellular networks,” in *IEEE Transactions on Communications*, vol. 57, no. 10, pp. 3059-3068, October 2009.

[52] A. Khanafer, W. Saad, T. Baar and M. Debbah, ”Competition in femtocell networks: Strategic access policies in the uplink,” *2012 IEEE International Conference on Communications (ICC)*, Ottawa, ON, 2012, pp. 5070-5074.

[53] L. Li, C. Xu and M. Tao, ”Resource Allocation in Open Access OFDMA Femtocell Networks,” in *IEEE Wireless Communications Letters*, vol. 1, no. 6, pp. 625-628, December 2012.

[54] H. A. Mahmoud and I. Gvenc, ”A comparative study of different deployment modes for femtocell networks,” *IEEE 20th International Symposium on Personal, Indoor and Mobile Radio Communications*, Tokyo, 2009, pp. 1-5.

[55] I. W. Mustika, K. Yamamoto, H. Murata and S. Yoshida, ”Potential Game Approach for Self-Organized Interference Management in Closed Access Femtocell Networks,” *Vehicular Technology Conference (VTC Spring)*, 2011 IEEE 73rd, Yokohama, 2011, pp. 1-5.

[56] G. de la Roche, A. Valcarce, D. Lopez-Perez and J. Zhang, ”Access control mechanisms for femtocells,” in *IEEE Communications Magazine*, vol. 48, no. 1, pp. 33-39, January 2010.

[57] J. Xiang, Y. Zhang, T. Skeie and L. Xie, ”Downlink Spectrum Sharing for Cognitive Radio Femtocell Networks,” in *IEEE Systems Journal*, vol. 4, no. 4, pp. 524-534, Dec. 2010.

[58] T. Zahir, K. Arshad, A. Nakata and K. Moessner, ”Interference Management in Femtocells,” in *IEEE Communications Surveys & Tutorials*, vol. 15, no. 1, pp. 293-311, First Quarter 2013.

[59] “ITU-R Guildlines for Evaluation of Radio Transmission Technologies for IMT-2000 RECOMMENDATION” ITU-R M.1225 1997, 26

[60] D. Hu and S. Mao, ”On Medium Grain Scalable Video Streaming over Femtocell Cognitive Radio Networks,” in *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 3, pp. 641-651, April 2012.

[61] D. Hu and S. Mao, ”Multicast in Femtocell Networks: A Successive Interference Cancellation Approach,” *Global Telecommunications Conference (GLOBECOM 2011)*, 2011 IEEE, Houston, TX, USA, 2011, pp. 1-6.

[62] Cormen, T. H.; Leiserson, C. E.; Rivest, R. L. and Stein, C. “Introduction to Algorithms, Third Edition” The MIT Press, 2009

[63] V. Chandrasekhar and J. G. Andrews, ”Uplink capacity and interference avoidance for two-tier femtocell networks,” in *IEEE Transactions on Wireless Communications*, vol. 8, no. 7, pp. 3498-3509, July 2009.

[64] M. Morita, Y. Matsunaga and K. Hamabe, ”Adaptive Power Level Setting of Femtocell Base Stations for Mitigating Interference with Macrocells,” *Vehicular Technology Conference Fall (VTC 2010-Fall)*, 2010 IEEE 72nd, Ottawa, ON, 2010, pp. 1-5.

[65] B. G. Choi, E. S. Cho, M. Y. Chung, K. y. Cheon and A. S. Park, ”A femtocell power control scheme to mitigate interference using listening TDD frame,” *The International Conference on Information Networking 2011 (ICOIN2011)*, Barcelona, 2011, pp. 241-244.

[66] K. Lee, S. Kim, S. Lee and J. Ma, ”Load balancing with transmission power control in femtocell networks,” *Advanced Communication Technology (ICACT)*, 2011 13th International Conference on, Seoul, 2011, pp. 519-522.

[67] Mostafa Zaman Chowdhury, M. T. Bui and Yeong Min Jang, ”Neighbor cell list optimization for femtocell-to-femtocell Handover in dense femtocellular networks,” 2011 *Third International Conference on Ubiquitous and Future Networks (ICUFN)*, Dalian, 2011, pp. 241-245.

[68] M. Z. Chowdhury, Won Ryu, Eunjun Rhee and Yeong Min Jang, "Handover between macrocell and femtocell for UMTS based networks," *11th International Conference on Advanced Communication Technology*, Phoenix Park, 2009, pp. 237-241.

[69] F. Pantisano, M. Bennis, W. Saad, M. Debbah and M. Latva-aho, "Interference Alignment for Cooperative Femtocell Networks: A Game-Theoretic Approach," in *IEEE Transactions on Mobile Computing*, vol. 12, no. 11, pp. 2233-2246, Nov. 2013.

[70] C. H. Chai, Y. Y. Shih and A. C. Pang, "A spectrum-sharing rewarding framework for co-channel hybrid access femtocell networks," *INFOCOM, 2013 Proceedings IEEE*, Turin, 2013, pp. 565-569.

[71] X. Ge et al., "Spectrum and Energy Efficiency Evaluation of Two-Tier Femtocell Networks With Partially Open Channels," in IEEE *Transactions on Vehicular Technology*, vol. 63, no. 3, pp. 1306-1319, March 2014. doi: 10.1109/TVT.2013.2292084

[72] Y. S. Liang, W. H. Chung, G. K. Ni, I. Y. Chen, H. Zhang and S. Y. Kuo, "Resource Allocation with Interference Avoidance in OFDMA Femtocell Networks," in *IEEE Transactions on Vehicular Technology*, vol. 61, no. 5, pp. 2243-2255, Jun 2012.

[73] H. S. Jo, C. Mun, J. Moon and J. G. Yook, "Self-Optimized Coverage Coordination in Femtocell Networks," in *IEEE Transactions on Wireless Communications*, vol. 9, no. 10, pp. 2977-2982, October 2010.

[74] S. E. Nai and T. Q. S. Quek, "Coexistence in two-tier femtocell networks: Cognition and optimization," *2012 International Conference on Computing, Networking and Communications (ICNC)*, Maui, HI, 2012, pp. 655-659.

[75] N. D. El-Din, E. A. Sourour, I. A. Ghaleb and K. G. Seddik, "Femtocells interference avoidance using Femtocell Identification," *Radio Science Conference (NRSC), 2011 28th National, Cairo*, 2011, pp. 1-9.

[76] W. C. Cheung, T. Q. S. Quek and M. Kountouris, "Spectrum allocation and optimization in femtocell networks," *2012 IEEE International Conference on Communications (ICC)*, Ottawa, ON, 2012, pp. 2473-2478.

[77] J. S. Lin and K. T. Feng, "Game Theoretical Model and Existence of Win-Win Situation for Femtocell Networks," *2011 IEEE International Conference on Communications (ICC)*, Kyoto, 2011, pp. 1-5.

[78] Y. Kim, S. Lee and D. Hong, "Performance Analysis of Two-Tier Femtocell Networks with Outage Constraints," in *IEEE Transactions on Wireless Communications*, vol. 9, no. 9, pp. 2695-2700, September 2010.

[79] Y. Sun, R. P. Jover and X. Wang, "Uplink Interference Mitigation for OFDMA Femtocell Networks," in *IEEE Transactions on Wireless Communications*, vol. 11, no. 2, pp. 614-625, February 2012.

[80] W. Zheng, T. Su, W. Li, Z. Lu and X. Wen, ”Distributed energy-efficient power optimization in two-tier femtocell networks,” *2012 IEEE International Conference on Communications (ICC)*, Ottawa, ON, 2012, pp. 5767-5771.

[81] C. Godsil and G.F. Royle, ”Algebraic Graph Theory”, *Springer-Verlag New York*, 2001

[82] Qualcomm, Making the Best Use of Unlicensed Spectrum for 1000x, May, 2015

[83] Huawei, U-LTE: Unlicensed Spectrum Utilization of LTE *Techincal Report*, [online] Available: http://www.huawei.com/ilink/en/download/hw_327803, 2014,

[84] T. Chen, Licensed Assisted Access: Operation Principles, *Ericsson Research Blog*, Feb. 2015. [online] Available: http://www.ericsson.com/research-blog/lte/license-assisted-access/,

[85] 3GPP, ”Further advancements for E-UTRA physical layer aspects, V9.0.0”, *TR 36.814* Mar. 2010

[86] M. J. Neely, ”Opportunistic scheduling with worst case delay guarantees in single and multi-hop networks,” INFOCOM, 2011 Proceedings IEEE, Shanghai, 2011, pp. 1728-1736.

[87] Qualcomm, ”Extending the benefits of LTE Advanced to unlicensed spectrum”, *Technical Report*, Apr. 2014

[88] Qualcomm, Making the Best Use of Unlicensed Spectrum for 1000x, *Technical Report*, [online] Available: https://www.qualcomm.com/media/documents/files/making-the-best-use-of-unlicensed-spectrum-presentation.pdf, May 2015

[89] Weber, Robert J., ”Auction Theory: By Vijay Krishna”, *Games and Economic Behavior*, vol. 45, pp. 488–497, Nov. 2003

[90] R. Zhang, M. Wang, L. X. Cai, Z. Zheng, X. Shen and L. L. Xie, ”LTE-unlicensed: the future of spectrum aggregation for cellular networks,” in *IEEE Wireless Communications*, vol. 22, no. 3, pp. 150-159, June 2015.

[91] Y. Wu, B. Wang, K. J. Ray Liu and T. C. Clancy, ”A scalable collusion-resistant multi-winner cognitive spectrum auction game,” in *IEEE Transactions on Communications*, vol. 57, no. 12, pp. 3805-3816, December 2009.

[92] C. Chen, R. Ratasuk and A. Ghosh, ”Downlink Performance Analysis of LTE and WiFi Coexistence in Unlicensed Bands with a Simple Listen-Before-Talk Scheme,” *2015 IEEE 81st Vehicular Technology Conference (VTC Spring)*, Glasgow, 2015, pp. 1-5.

[93] Cavalcante, A.M. and Almeida, E. and Vieira, R.D. and Chaves, F. and Paiva, R.C.D. and Abinader, F. and Choudhury, S. and Tuomaala, E. and Doppler, K., Proc. IEEE VTC-Spring 2013, *Performance Evaluation of LTE and Wi-Fi Coexistence in Unlicensed Bands*, June, 2013, Dresden, Germany, pp. 1–6

[94] Nihtila, T. and Tykhomyrov, V. and Alanen, O. and Uusitalo, M.A. and Sorri, A. and Moisio, M. and Iraji, S. and Ratasuk, R. and Mangalvedhe, N., Proc. WCNC 2015, *System performance of LTE and IEEE 802.11 coexisting on a shared frequency band*, Apr., 2013, ew Orleans, LA, pp. 1038–1043

[95] Ratasuk, R. and Mangalvedhe, N. and Ghosh, A., *LTE in unlicensed spectrum using licensed-assisted access* Dec., 2014, Austin, TX, pp. 746–751

[96] A. Al-Dulaimi, S. Al-Rubaye, Q. Ni and E. Sousa, "5G Communications Race: Pursuit of More Capacity Triggers LTE in Unlicensed Band," in *IEEE Vehicular Technology Magazine*, vol. 10, no. 1, pp. 43-51, March 2015.

[97] A. Bhorkar, C. Ibars and P. Zong, "On the throughput analysis of LTE and WiFi in unlicensed band," *2014 48th Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, CA, 2014, pp. 1309-1313.

[98] F. Teng, D. Guo and M. L. Honig, "Sharing of unlicensed spectrum by strategic operators," *Signal and Information Processing (GlobalSIP)*, 2014 IEEE Global Conference on, Atlanta, GA, 2014, pp. 288-292.

[99] B. Gao, Y. Yang and J. M. J. Park, "A credit-token-based spectrum etiquette framework for coexistence of heterogeneous cognitive radio networks," *IEEE INFOCOM 2014 - IEEE Conference on Computer Communications*, Toronto, ON, 2014, pp. 2715-2723.

[100] Juncheng Jia, Qian Zhang, Qin Zhang, and Mingyan Liu. 2009. Revenue generation for truthful spectrum auction in dynamic spectrum access. In *Proceedings of the tenth ACM international symposium on Mobile ad hoc networking and computing (MobiHoc '09)*. ACM, New York, NY, USA, 3-12.

[101] H. Li, C. Wu and Z. Li, "Socially-optimal online spectrum auctions for secondary wireless communication," *2015 IEEE Conference on Computer Communications (INFOCOM)*, Kowloon, 2015, pp. 2047-2055.

[102] A. Bhorkar, C. Ibars, A. Papathanassiou and P. Zong, "Medium access design for LTE in unlicensed band," *Wireless Communications and Networking Conference Workshops (WCNCW)*, 2015 IEEE, New Orleans, LA, 2015, pp. 369-373.

[103] A. M. Voicu, L. Simi and M. Petrova, "Coexistence of pico- and femto-cellular LTE-unlicensed with legacy indoor Wi-Fi deployments," *2015 IEEE International Conference on Communication Workshop (ICCW)*, London, 2015, pp. 2294-2300.

[104] A. Al-Dulaimi, S. Al-Rubaye, Q. Ni and E. Sousa, "5G Communications Race: Pursuit of More Capacity Triggers LTE in Unlicensed Band," in *IEEE Vehicular Technology Magazine*, vol. 10, no. 1, pp. 43-51, March 2015

[105] Z. Jiang and S. Mao, "Energy Delay Tradeoff in Cloud Offloading for Multi-Core Mobile Devices," in *IEEE Access*, vol. 3, no. , pp. 2306-2316, 2015.

[106] T. Schwengler and M. Glbert, ”Propagation models at 5.8 GHz-path loss and building penetration,” *Radio and Wireless Conference, 2000. RAWCON 2000*. 2000 IEEE, Denver, CO, 2000, pp. 119-124.

[107] P. Gupta and P. R. Kumar, ”The capacity of wireless networks,” in *IEEE Transactions on Information Theory*, vol. 46, no. 2, pp. 388-404, Mar 2000.

[108] Lawrence M. Ausubel and Paul Milgrom, ”The Lovely but Lonely Vickrey Auction”, *Combinatorial Auctions, chapter 1,* MIT Press, 2006