Dynamic Spectrum Decision in Multi-channel Cognitive Radio Networks with Heterogeneous Services

by

Hongqiao Tian

A Thesis submitted to the Faculty of Graduate Studies of
The University of Manitoba
in Partial Fulfilment of the Requirements for the degree of

Master of Science

Department of Electrical and Computer Engineering
University of Manitoba
Winnipeg

Abstract

With the fast growing of wireless communication applications, cognitive radio has become popular in recent years as an effective dynamic spectrum access technique to improve spectrum utilization efficiency. However, the channels that are allocated to the secondary users (SUs) can be re-occupied by the primary users (PUs) at any time which makes it difficult to meet SUs' quality of services (QoS) requirements. Therefore, the SUs may simultaneously use multiple noncontiguous spectrum bands for transmission. It is important to study how to make spectrum decisions for the SUs in multi-channel cognitive radio neworks (CRNs) to meet their heterogeneous QoS requirements. In this thesis, after presenting some fundamentals and related works, we discuss a dynamic load-balancing spectrum decision framework where the SU with prioritized services dynamically selects the most suitable operating channels for packet transmission. A discrete time priority queueing model is applied to model multiple interruptions from PUs and heterogeneous channel conditions. Optimal spectrum decision policies are obtained to achieve minimum delay using dynamic programming techniques, such as Markov decision process (MDP) and reinforcement learning, under different assumptions. To address the computational complexity issue in the MDP solutions, a myopic scheme is proposed based on the estimated packet sojourn time. Simulation results demonstrate the effectiveness of all proposed algorithms for load-balancing spectrum decision. It also shows that the proposed myopic scheme can achieve significant reduction on computational complexity with a cost on the delay performance of low priority BE services.

Acknowledgements

Immeasurable appreciation and deepest gratitude for the help and support are extended to the following persons who in one way or another have contributed in my study and made this dissertation possible.

First and foremost I offer my sincerest gratitude to my advisor, Dr. Jun Cai, for providing me the opportunity of taking part in this Master program, for his valuable guidance, encouragement, professionalism and gracious support throughout my entire program of study. He allowed me the room to work in my own way, but steered me in the right direction whenever he thought I need it. I am also thankful to him for carefully reading and commenting on countless revisions of this manuscript. I do not have enough word to express my deep and sincere appreciation.

I am hugely indebted to Professor Attahiru S. Alfa for being ever so kind to show interest in my research and giving me his precious and kind advice regarding the topic of my research. I am deeply grateful to him for finding time for me in his busy schedule to help me sort out the technical details of my work, and providing me with materials I could not possible to discover on my own. I would also like to thank my committee member, Professor Wang, for reading the manuscript and his insightful suggestions and valuable comments on my thesis.

I am also grateful to my fellow labmates in the Robert Alan Kennedy Communication Lab for providing a simulating and fun environment to learn and grow. I would like to acknowledge Shiwei Huang, Huijin Cao, Changyan Yi, and Chamara N. Devanarayana for many valuable discussions that helped me understand my research area better. I wish to

thank Kamal Darchini for patiently helping me in queueing theory and Zheng Zhao for his help in overcoming difficulties in simulations. My sincere thanks to Ehsan Montazeri for reviewing my manuscript and support through the difficult times.

Lastly and most importantly, none of this would have been possible without the love and patience of my family. I would like to express my heart-felt gratitude to my parents and my grandma for supporting me spiritually throughout my life.

Contents

List of Figures

Chapter 1

Introduction

1.1 Background and Motivations

Radio spectrum is used for wireless communication which ranges from 3KHz-300GHz. From cellphones to security systems, from television remote control to wireless headphones, virtually every wireless device depends on access to the radio frequency (RF) wireless spectrum. Each application / service / user needs a certain bandwidth of spectrum. For example, the bandwidths for WiFi, WCDMA and TV are 20MHz, 5MHz and 6MHz respectively. Radio spectrum is one of the most tightly regulated resources of all the time. To ensure interference-free communications between users, since 1930s, fixed portions of wireless spectrum were assigned by governmental agencies to licensed holders based on long term policies.

Recent years, demand for wireless band has increased rapidly due to technology development, such as 3G and 4G, and the rapid expansion of wireless internet services. The radio spectrum has therefore become a scarce commodity in many countries. However, according to a report published by Federal Communications Commission (FCC) in 2002, spectrum access is a more significant problem than physical scarcity of the spectrum [1]. The reason is that exclusive access through licensing limits the ability of potential users to obtain access, leading to underutilization of a significant amount of spectrum. According to the report, the licensed spectrum are utilized 15 to 85 percent with a high variance at different geographic locations at a given time. To keep up with the demand, as an alternative to exclusive access, dynamic spectrum access (DSA) techniques were proposed to solve these spectrum inefficiency problems [2].

Cognitive radio (CR) is a key enabling technology for efficient DSA which provides the capability to access the usable spectrum opportunistically and dynamically [3] [4]. CR is an intelligent radio that can monitor, sense and detect the available channels from surrounding wireless spectrum, and dynamically change transmission parameters, enabling more communication to run concurrently and also improving radio operating behavior. By opportunistic access to the idle channels, cognitive users (also called as secondary users (SUs)) can share the spectrum and extract more bandwidth which improves overall spectrum efficiency. The FCC and wireless regulatory bodies around the world are in the process of opening up new spectrum, as well as reclassifying the existing spectrum, to make it available for opportunistic use for CRs. The FCC and the Office of Communications in the United Kingdom have opened up some unused portions of the RF spectrum (known as white spaces) for public use in 2008 and 2010, respectively [5] [6]. This would allow new market entrants, utilities, public safety, enterprise and even existing wireless operators to offer new services with additional bandwidth and higher capacity without requiring these entities to purchase expensive and scarce wireless spectrum.

Cognitive radio networks (CRNs) have some challenging characteristics, such as high fluctuation in the available spectrum and diverse QoS requirements of various applications. To realize efficient spectrum utilization using CR technology, a dynamic spectrum management framework is required. The spectrum management framework consists of spectrum sensing, spectrum decision, spectrum sharing and spectrum mobility [7]. Through spectrum sensing, SUs detect available channels that are not occupied by the licensed users, also called as primary users (PUs). Spectrum enables one SU access the channel coordinately with other SUs. With spectrum mobility, the SUs are able to vacate the channel when the PU is detected. Spectrum decision is an important part of spectrum management which helps SUs to select the best available channels for transmission to satisfy their QoS requirements. We will focus on the component of spectrum decision process in this work.

During the spectrum decision process, the vacant spectrum bands are firstly identified and each spectrum band is characterized based on the observed and statistical information about the PU's activity. After the available spectrum bands are characterized, the next step is to select the most suitable spectrum bands. In the last step, the CR transmitter reconfigures its transmission parameters to support best services for the SU while keeping interference on the PUs under certain limits. Due to the characteristics of unstable topology and time varying RF propagation of CRNs, spectrum decision is mostly proceeded in a dynamic manner. A comprehensive literature review of spectrum decision are presented in Chapter 2.

The motivation of this thesis is mainly concentrated on the following.

Most works in the literature review assume that SUs support only single class traffic. However, in practical applications, SUs may need to support different classes of services, such as delay sensitive (DS) services (video conference and voice over IP), and best effort (BE) services (file transfer and video streaming) [8]. In most of the works that considered heterogenous traffic, each SU or each connection is assigned to only one channel at the same time. But to achieve higher data rate in multi-channel CRNs, the SUs with multi-radios will be able to transmit data through multiple channels simultaneously. How to allocate packets with different QoS requirements to multiple channels will be a challenging problem. A few works have proposed some channel selection policies regarding the case with multi-channel transmission. They focused on the problem that how many channels should be assembled or reserved by the high priority SU and the proposed policy are based on intuitive ideas. However, no optimal dynamic channel selection policy with the consideration of both RF environment characterization and PUs' activities has been proposed. To address this channel selection issue, in chapter 3, we proposed a dynamic load-balancing spectrum decision scheme for the SU in multi-channel CRNs with multi-class services. With the proposed scheme the SU's traffic load can be distributed effectively to the channels, thus to improve overall delay performance. Further more, heterogeneous PUs' activities, channel characteristics and buffer dynamics are considered when finding the optimal channel selection policies.

1.2 Summary of Contributions

The contribution of this thesis are summarized as follows:

The dynamic spectrum decision process is addressed in a CRN consisting of multiple PUs and one SU which support both DS and BE services. The SU is given the opportunity to access multiple channels shared by the two services. To avoid the head of line effect [9–11], we consider a system model where the SU maintains one buffer at each channel to buffer the interrupted packets or the packets waiting to be transmitted. To analyze the transmission delay of the SU's traffic, a priority queueing model with ON/OFF server is developed. Based on the queueing analysis, Markov decision processes (MDPs) are formulated to find the optimal channel selection policies, according to which the SU's DS and BE packets are distributed to all the available channels to minimize their average delay. To evaluate the average delay ,we jointly consider heterogeneity in PU activities, channel service rates and number of packets in each buffer. An optimal policy is obtained using relative value iteration. We further extend our work to the case where the distributions of packet arrivals at the primary transmitters and the channel statistics of the primary and secondary links are unknown. A form of reinforcement learning (RL) technique, which is known as R-learning, is introduced to solve the MDPs without a priori knowledge of the state transition probabilities. Moreover, to reduce the huge complexity in the solution to the MDP problem, a myopic method with low computational complexity is also proposed where the decisions are made to minimize the immediate cost, defined as the expected delay of the arriving packet.

1.3 Outline of the Thesis

The rest of the thesis is organized as follows. Chapter 2 introduces some fundamentals and related works that are relevant to our research. Motivated by the limits of existing spectrum decision techniques in multi-channel CRNs, a dynamic load-balancing spectrum decision scheme is proposed and the policies obtained via different techniques are evaluated through simulations in Chapter 3. Chapter 4 presents a brief conclusion of this thesis and summarizes some possible extensions as future works.

Chapter 2

Fundamentals and Related Works

In this chapter, fundamental knowledge and related literatures are presented as the basis for future reference. We first provide an overview of CRNs including their architectures, spectrum decision framework and research challenges. Channel selection, as a major issue in spectrum decision, has been widely discussed in CRNs. A comprehensive literature review about channel selection is therefore presented. After that, some basic knowledge of the MDP and RL is introduced.

2.1 Overview of Cognitive Radio Networks

The concept of CR was first proposed by Joseph Mitola III in 1998 and later published in an article in 1999. It is defined as "*an intelligent wireless communication system capable of changing its transceiver parameters based on interaction with the external environment in which it operates*" [12]. Conventional radios are designed under the assumption that they were operated in interference-free spectrum and therefore unable to dynamically change transmission parameters, switch between different channels. Different from conventional radios, CRs are able to monitor and sense their operating condition and dynamically reconfigure their transmission parameters, such as power output, frequency and modulation, to best match the environment. Therefore, an optimized communication experience for user can be ensured.

2.1.1 Cognitive Radio Technology

The CRNs can be classified as two groups: a primary network and a secondary network [13]. The spectrum bands are licensed to the users in the primary network and therefore the PUs have higher priority in spectrum access and their operation should not be affected by the SUs. The secondary network (also called CR network or unlicensed network) does not have a license to access the primary network's spectrum band. Hence, additional functions are required for the SUs. For example, they have to detect if PUs are occupying the channels or not before access to the spectrum. Furthermore, if PUs appear in the spectrum band occupied by SUs, the SUs should vacate the spectrum band immediately and switch to other available channels.

To achieve these functions, CR has two main characteristics which can be defined as [14]:

Cognitive capability: By analyzing the radio environment SUs can detect spectrum holes, which are defined as a band of frequencies assigned to a PU, but at a particular time and specific geographic location, the band is not being utilized by that user [15]. It can be achieved by direct spectrum sensing, using geo-location database, beaconing techniques or the combination of spectrum sensing and geo-location database [16]. SUs can also estimate the channel state information and predict the channel capacity.

Reconfigurability: CR is built on software-defined radio which can operate on different frequency bands and access channels with various techniques. With this capability, SUs can accommodate new interface standards and exploit heterogenous applications and services.

CRNs impose unique challenges because of the coexistence with the primary network and diverse QoS requirements of different applications. First of all, when sharing spectrum with primary networks, the interference on PUs should be limited. Second, QoS-aware communication needs to be supported under the dynamic and heterogenous spectrum environment. Finally, seamless communication should be ensured regardless of the appearance of PUs. Therefore, new spectrum management techniques are required to deal with these critical challenges.

To address these challenges, spectrum management in CRNs consists of the following steps as shown in Fig. 2.1 [7]:

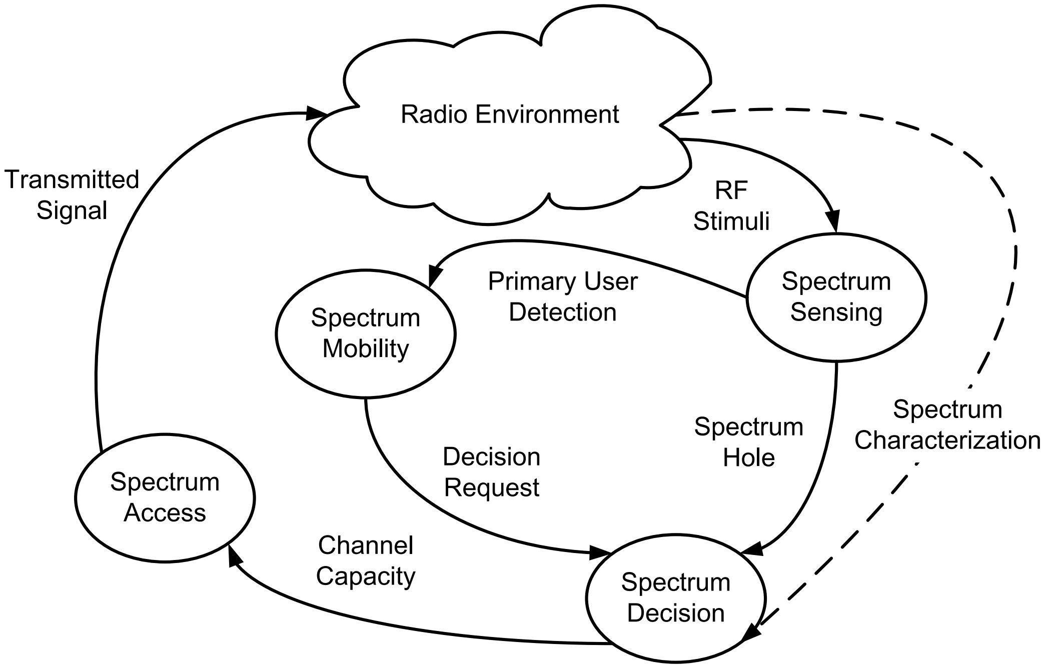

Figure 2.1: Dynamic Spectrum Management Framework

Spectrum sensing: To access the unused spectrum band, an SU should monitor available spectrum bands and collect channel information to detect spectrum holes through spectrum sensing.

Spectrum decision: Among the detected available spectrum bands, the SU should select the best operating channel based on channel characteristics and QoS requirements.

Spectrum sharing: For the cases where multiple SUs try to access the same spectrum bands, the SUs should be able to coordinate with each other to avoid collision.

Spectrum mobility: Since SUs have lower priority than PUs, multiple interruptions by PUs may occur during each connection. Therefore, to achieve seamless connection, the SU must be able to continue transmission at other vacant channels.

With all these functions, CR is able to exploit spectrum access opportunities efficiently. However, the heterogeneous spectrum environment makes spectrum decision a critical issue for CRNs. Generally, multiple available spectrum bands may be found over a wide frequency range that have different characteristics and the CRNs need to support different applications. Through spectrum decision process, CRs select the best spectrum band according to the application requirements once available spectrum bands are identified.

2.1.2 Spectrum Decision Framework for CRNs

Spectrum decision involves three main functions: spectrum characterization, spectrum selection and CR reconfiguration [7]. Once available spectrum bands are identified through spectrum sensing, each spectrum band is characterized based on radio conditions and the PU's activity. In the second step, the most appropriate spectrum band is selected according to a predefined decision making policy which is obtained based on the spectrum band characterization. Third, a CR should be able to reconfigure transmission parameters to support communication on the selected band. The relationship of required functions for the spectrum decision framework is described in Fig. 2.2 [17].

A. Spectrum Characterization

SUs characterize the spectrum band by considering the received signal strength, interference and the number of users currently operating on the spectrum. SUs should also observe PUs' activities which cause spectrum holes fluctuating over time and location. Spectrum characterization should include RF environment characterization and PU activity modelling [17].

1) Radio Frequency Environment Characterization

Since available spectrum bands have various characteristics, channel characterization is continuously performed in CRNs. RF environment characterization involves channel identification and estimation of channel capacity.

Channel identification includes environment learning and primary traffic classification. As CRNs can be applied to different networks, such as television white space networks,

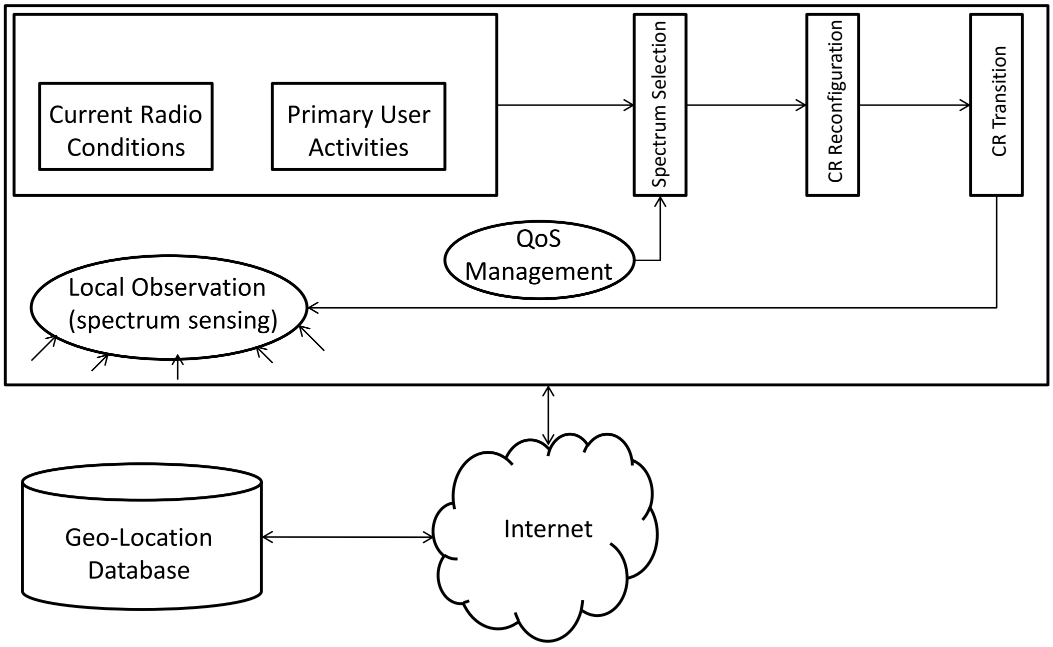

Figure 2.2: Spectrum Decision Framework

smart grid networks, machine-to-machine networks, public safety networks, broadband cellular networks and wireless medical networks [18], the primary network traffic patterns can be either deterministic or statistic. In the networks with deterministic traffic, such as TV broadcasting, the channels are occupied by PUs at fixed time slots. Once the PUs stop communicating, the channels become available for the SUs. Therefore, fixed channel idle periods make it easier to predict the future channel states based on the past observed values. On the other hand, for the networks with stochastic traffic, such as cellular networks, PUs' activities vary in time and space which can only be predicted using probabilities and statistics. A prediction method for both deterministic and stochastic traffic patterns is proposed in [19].

To estimate channel capacity in CRNs, some factors need to be considered such as channel interference, switching delay and holding time.

- *Channel interference*: The SUs may lead harmful interference on PUs operating on adjacent channels. It is important to estimate channel interference on the PUs caused by CRNs which can be controlled by limiting the transmission power.

- *Switching delay*: In CRNs, each SU can have multiple available channels to perform opportunistic access. Switching delay is caused when the SU switches from one channel to another, which happens when the PU is detected on the operating channel or the degradation of QoS. During the switching process, transmissions between transmitters and receivers are temporally suspended until the communication is resumed on another idle channel.

- *Channel holding time*: Channel holding time is the expected duration that a SU can occupy the channel before PU's interruption [7]. The longer holding time the channel has, the better QoS experience the SU can achieve.

2) Primary User Activity Modelling

CRNs access the spectrum band when PUs are not utilizing it. However, it cannot be ensured that the spectrum band will be idle during the entire SU's communication period. The transmission will be interrupted once the PU appears, and switching delay or packet loss will be incurred. Therefore, it is very important to predict the PU's activity, which is defined as the probability of the PU's appearance during SU's transmission, based on the history of spectrum usage information. The process is called PU activity Modelling. According to PU activity modelling, the CRNs can be aware of spectrum fluctuation which is an important factor for making spectrum decision. With appropriate PU activity modelling, CRNs can utilize spectrum effectively.

The most frequently used PU activity modelling method is Poisson Modelling [20], where the PU activity is modelled as a Poisson process with exponentially distributed inter-arrivals. The PU traffic is modelled as a two-state birth-death process where birth rate denotes the PU's arrival rate and death rate denotes the transmission rate. Let ON and OFF represent the channel is idle or not, respectively. As a result, the durations of ON and OFF are exponentially distributed. Some papers [19, 20] proposed some modelling techniques based on statistics. In paper [19], primary traffic was classified into deterministic and stochastic traffic based on stored spectrum usage information. Channel selection decision for the SU was made based on predicated channel idle time. Paper [20] characterized a channel based on the statistics of the TV channels which were classified into long-term and short-term usage.

B. Spectrum Selection

After available spectrum bands are characterized, the next important step is to select the best spectrum bands among them according to the SU's QoS requirements. According to different QoS measurements, the decision schemes can be divided into two main groups: delay minimum schemes and throughput maximum schemes. Different approaches have been applied in each group.

Some media access control (MAC) protocols based on partially observed Markov decision process (POMDP) have been proposed to find the optimal channels to sense and access in order to achieve optimal opportunistic spectrum access [21–23]. Game theoretic frameworks [24, 25] were proposed to solve channel selection problems. Based on the game theory models, each SU decides the best channel selection probabilities to maximize its utility function. Joint channel and power allocation optimization problems have been formulated in some papers to maximize the total throughput under QoS and interference constraints [26, 27]. Some policy-based spectrum selection schemes were proposed, such as sequential selection and weighted selection [28–30]. In sequential selection schemes, the available channels are ordered according to a pre-defined policy and the channel selection procedure continues until there are no more idle channels. In weighted selection, weights are given to each selection criterion and the best channels are selected based on the sum of weighted values. For delay performance analysis and practical considerations, priority queueing theory is introduced in CRNs to model the priority of PUs over SUs, as well as multiple interruptions from the PUs. A detail channel selection literature review will be presented in the following section.

C. Reconfiguration in CRNs

In traditional wireless network, users operate on pre-defined frequency bands with pre-defined transmitter parameters. The existing hard-ware based architecture limits the flexibility to adapt to the external environment. To achieve heterogeneous spectrum ability and dynamic spectrum access, CRNs are implemented on software defined radio (SDR) to rapidly adjust their transceiver parameters based on the external RF environment, policy updates, QoS requirements, selected spectrum, channel characteristics and the needs of the users. Reconfiguration of parameters occurs after the operating channel is characterized and selected.

To adapt to the QoS requirements and regulatory policies, the following reconfiguration parameters are mainly included.

- *Modulation and Coding Schemes*: Reconfiguration in modulation and coding is needed when the SU's QoS requirements or channel conditions are changed. An adaptive transmission scheme was proposed in [31] which adaptively selected the modulation order that can achieve the maximum throughput for the SUs.

- *Transmission Power*: Power control is an important issue in CRNs which has been discussed in the literature [32, 33]. The objective of power control is to support the QoS requirement while minimizing energy consumption and limiting interference to PUs and other SUs.

- *Operating Frequency*: Operating frequency reconfiguration is another key capability of CRNs which enables SUs to dynamically adapt to the RF environment. A predictive model is proposed in [34] to dynamically select the correct configurations including operating frequency.

- *Channel Bandwidth*: To transmit data on heterogeneous networks SUs has to be able to support variable channel bandwidths. For example, if a SU intends to utilize both TV white spaces and 2.4G WiFi spectrum bands, it has to adapt its channel bandwidth to 20MHz, 40MHz, 5MHz and 10 MHz.

- *Communication Technology*: CRNs are heterogenous wireless networks that can intemperate with different communication systems such as GSM, WiFi and LTE. Therefore, it is necessary for a SU to be able to use different communication technologies.

2.1.3 Research Challenges in Cognitive Radio Networks

In this section, the main challenges in CRNs will be discussed.

1. Spectrum Sensing

Existing spectrum sensing techniques mainly include energy detection and feature detection. For energy detection, since the energy detector is not able to distinguish signal types, false alarm may happen when uncertainty noise power is mistakenly considered. Although feature detection is robust, complex computation and long observation time are required. Since longer sensing time leads to shorter transmitting time, inefficient sensing can possibly leads to performance degradation. Therefore, it is a challenge to design a sensing technique which can accurately detect PU signals with low computational complexity.

2. Spectrum decision in Heterogeneous Traffic Networks

Because of the low priority in utilizing the spectrum band, SUs cannot obtain a reliable channel for a long period of time and may not detect any single channel to meet their QoS requirements. Therefore, multiple noncontiguous spectrum bands can be used simultaneously by the SUs for transmission as shown in Fig. 2.3 [4]. With multi-channel transmission, the SUs can not only achieve higher throughput but also more reliable transmission. Even if the transmission on one channel is interrupted, the rest of the channels can still maintain communication. A major challenge is to select the best channels in multi-channel CRNs to meet the SU's heterogenous QoS requirements.

3. Channel Selection in Multi-hop CRNs

In multi-hop CRNs, each transmission is completed via multiple nodes. Each relay node receives packets on one channel and then transmits them on another channel which

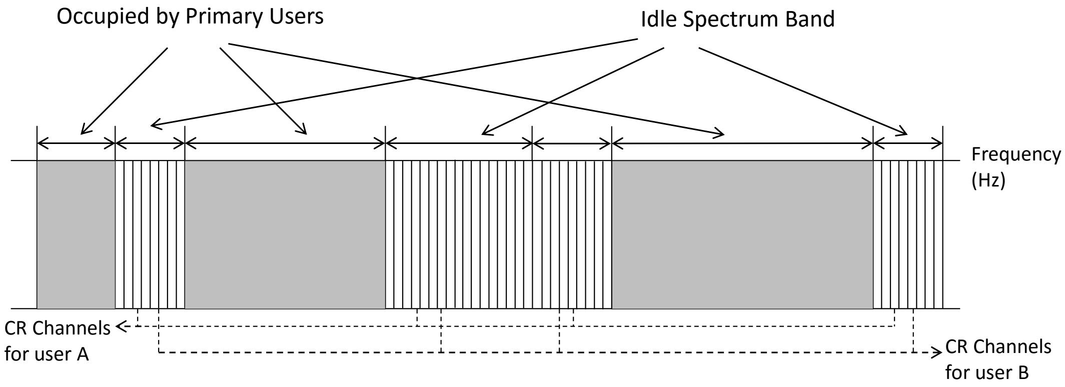

Figure 2.3: Channel structure of the multi-spectrum decision.

leads to cumulative delay. Therefore, to minimize delay along the path it is necessary to develop a joint spectrum and route selection approach.

4. Economic incentive and rationality

To opportunistic access the licensed primary networks, it is important to provide sufficient economic incentive for PUs to participate in spectrum sharing. Therefore, it is a challenge to balance economic rationality and fairness in CRNs.

5. Transmission security

Network security is an important issue in wireless networks. CRNs are built based on intelligent radio which leads to higher probability for potential attacks [35]. It is a critical issue to provide secure transmissions for both SUs and PUs.

2.2 Related Works in Channel Selection

To select the most suitable spectrum for SUs' heterogeneous QoS requirements, many factors, such as spectrum sensing results, PU modelling, channel characteristics, network topology and media access control (MAC) need to be considered. Due to the importance and complexity, channel selection is one of the most popular topics in CRNs. In this section, a general survey on channel selection techniques is presented.

Channel selection techniques can be classified by different aspects. Based on the objective, it can be mainly divided into throughput maximization and delay minimization. It can also be grouped according to the network structures, i.e., centralized and decentralized CRNs. Based on the service supported, there are channel selections for CRNs with multi-class or single-class services. It can also be categorized into connection based or packet-wise channel selections, which depends on when the channel selection is executed. For connection based channel selection, it happens when a connection is interrupted. For packet-wise channel selection, it happens in each slot. Here, we classify the existing channel selection approaches into channel selection with single class and multi-class services.

A. Channel selection for Secondary Users with Single class Services

In the CRNs with single class traffic, all the SUs have the same priority in selecting operating channels. This problem has been studied in different settings within the literature [21–23] [36–44].

Some papers have proposed cross-layer approaches, which integrate spectrum sensing and access, to achieve opportunistic spectrum access. In paper [21], Zhao et al. proposed decentralized MAC protocols that allowed SUs to choose a set of channels to sense and select a set of channels to access at the beginning of each time slot without a central coordinator. They developed an analytical framework for opportunistic spectrum access based on the theory of Partially Observable Markov Decision Process (POMDP). Under this framework, an SU made optimal decisions for sensing and access based on the belief vector that summarized the knowledge of channel states based on all the past decisions and observations. In the formulation, sensing errors and collisions were also considered in limiting the interference perceived by PUs. To reduce complexity, a suboptimal strategy with comparable performance was also developed. In paper [22], the authors considered the scenario where the at the beginning of each slot, an SU selected one channel to sense, and access if the channel was sensed to be at good state. An POMDP problem was formed to obtain the best channel selection policy with the objective of maximizing the throughput. They have established the structure of myopic policy for designing sensing strategy with low complexity, analyzed the performance and partly obtained the optimality for the case of independent and identically distributed (i.i.d) channels. Later in [23] and [36], the optimality of the myopic policy was derived for access to only one channel and multiple i.i.d channels each time, respectively, with positively correlated i.i.d channels.

Some works addressed joint spectrum allocation and power control for the SUs. Digham in paper [37] studied a CRN where a set of channels were assigned among multiple SUs that opportunistically accessed to the available spectrum. They jointly considered channel and power allocation while maximizing the total throughput of the CRN under interference constraints on PUs. The optimization problem was solved in a modified form of water filling. In [38], a CRN under the microscopic spectrum opportunity setting was explored, where a same channel might simultaneously present different levels of availability to different CRs. To coordinate channel access between SUs and ensure efficient utilization of spectrum opportunities, they formulated the joint power control and channel assignment problem as a mixed integer nonlinear programming problem (MINLP). The solution was obtained by transforming the MINLP into a binary linear program (BLP) that contained only binary variables and linear objective function and constraints. Both centralized and distributed algorithms, aiming at better performance and better implementability, were developed for the BLP, respectively. Q-learning was applied in [39] to solve channel and power allocation for the incoming service of a specific SU, where the arrival and departure of the SU's services were used to learn the optimal strategy to maximize the total system throughput. The proposed algorithm can be applied to centralized CRNs where each SU is constrained to transmit over at most one channel. A distributed joint spectrum allocation and power control strategy in multi-hop CRNs was also derived in [40]. Cooperation between nodes was introduced to deal with the interflow interference and cumulative interference so that multiple flows were able to coexist. Optimal waiting time was derived by balancing the tradeoff between spectrum efficiency and route switching overhead.

Channel selection involving buffer dynamics in a cognitive setting has been considered in a few works such as [41–44]. Wang et al. in [41] proposed a preemptive resume priority (PRP) M/G/1 queueing model to model the priority of PUs over SUs, as well as the multiple interruptions caused by PUs. In the model, each channel has two types of customers. The connections of the PUs and SUs join the high priority queue and the low priority queue, respectively, and the PUs have preemptive priority to interrupt the SUs' transmission. A probability-based load-balancing spectrum decision scheme was designed on top of the queueing model, where the SUs' traffic load was properly distributed to multiple channels. By deriving the optimal channel selection probabilities, the minimum system delay was achieved. Similarly, in [42], Do et al. proposed a lightweight algorithm to calculate channel selection probabilities and analyzed system delay based on an M/M/1 queueing model with a breakdown server. Based on a similar queueing model as [41], F. Sheikholeslami in [43] further considered the target channel selection policies after the occurrence of an interruption due to the arrival of a primary connection. They proposed a joint probabilistic approach for initial and target channel selection schemes for a SU in a CRN and analyzed the delay performances under different handoff policies. In [44], an orthogonal band allocation scheme was proposed where each user randomly accessed one band at the beginning of each time slot with a predefined probability and the stability region of the proposed system was analyzed.

All the above work considered scenarios where SUs support only one class of traffic, however, in practical applications, SUs may need to support different classes of services, such as delay sensitive (DS) applications (video conference and voice over IP), and best effort (BE) services (file transfer and video streaming). To address this issue, some spectrum decision schemes have been proposed by considering priority-based secondary users.

B. Channel selection for Secondary Users with Multi-class Services

Supporting SUs with multi-class services in CRN is challenging because of heterogenous QoS requirements. It is important to allocate wireless channel resources efficiently to ensure quality-driven transmissions.

To analyze the delay performance of SU with prioritized traffic, different queueing models have been proposed. In [45], a queueing-based dynamic channel selection scheme was developed for heterogeneous multimedia autonomous users. They proposed a priority virtual queue framework to consider different priorities of access to the channels and different channel conditions. In the framework, each user maintains M physical queues for various frequency channels and a "virtual queue" was formed at the same frequency channel. Based on the queueing model, expected utility was evaluated for each user at different frequency channels, according to which they proposed a decentralized learning algorithm that could dynamically adapt the channel selection strategies to maximize the utility functions. To manage and characterize spectrum usage behaviour of PUs and SUs in multimedia transmissions, Wu et al. in [46] proposed a mixed preemptive and non-preemptive resume priority (PRP/NPRP) M/G/1 queueing model. In the proposed model, a PRP M/G/1 queueing model was formed to ensure that the PUs had preemptive control over the channels and their transmission will not be effected by the SUs. The queueing among SUs was modelled as NPRP to avoid SUs from frequent spectrum handoffs due to the interruption from other SUs but also provide differentiated service. A reinforcement learning handoff scheme was proposed to adaptively perform spectrum handoff under changing channel conditions and traffic loads to maximize the transmission quality for the prioritized multimedia SUs. A virtual queue with different priorities was applied to model the traffic of PUs and SUs on the same channel [47]. To avoid starvation of delay insensitive SU's packets, a transmission window (TW) strategy was applied so that the packets of both delay sensitive SUs (DSP) and delay insensitive SUs (DIP) inside the TWs were first served under the condition that DSPs had priority over DIPs. Delay analysis under this strategy was conducted and a dynamic adaptive channel selection strategy based on learning automata was developed with the objective to reduce the queueing delay.

Some policy-based channel allocation schemes have been proposed to meet heterogeneous SUs QoS requirements. Jiang et al. in paper [29] established a QoE-driven channel allocations scheme where the sub-bands with smaller switch/dropping probabilities were allocated to the SUs with delay sensitive traffic to improve their performance. Similarly, in paper [30], channel holding time was considered as the principle for channel selection. The channels with more holding time were assigned to the applications with a higher priority so that less handoff would be needed. Balapuwaduge et al. in [8] proposed a queueing-based dynamic channel assembling strategy which introduced two queues dedicated for real-time SU (RSU) services and elastic SU (ESU) services, separately. In this strategy, both real time and elastic SUs could assemble several channels to increase their service rates, but the number of aggregated channels for real-time services was fixed. Two schemes were proposed. In the first scheme, when RSU services was interrupted and there was no available channels, the ESUs would be forced to terminate the service on one of the channels and denote it to the RSU. While it was not needed in scheme 2 if its minimum number of channels requirement would not be met. Therefore, the first scheme was more appropriate for delay-critical applications. Similarly, in [48], a certain number of channels were reserved for the high priority user and two dynamic channel access schemes with different handoff schemes were proposed. In the first handoff scheme, the ongoing low priority SU calls were terminated if the required idle channels were not available for high priority SU's handoff. While in the second scheme no ongoing low priority calls were terminated for the sake of the high priority SU's handoff. Based on the proposed schemes, performance analysis, such as blocking probabilities, forced termination probability and throughput, were derived. The optimal sub-channel reservation was also obtained.

Join power and channel allocation for heterogeneous services with imperfect sensing has been considered in papers [26] and [27]. In [26], the services was classified into ones with minimum-rate guarantee and the others with best effort services. An optimization problem was formulated with the objective to maximize the total capacity of CRNs under the total power constraint, minimum rate guarantee constraint and proportional-fairness constraint. An aggressive discrete stochastic approximate algorithm was proposed to reduce the computational complexity. Similar resource allocation optimization problem was formulated in [27], but taking mutual interference into consideration. To solve the problem, channel allocation was performed in the first step based on channel gains and interferences to PUs, and then power was allocated among the assigned channels.

2.3 Markov Decision Process and Reinforcement Learning

Since the system state dynamically evolves with time in CRNs, spectrum decision has to be proceeded dynamically. Markov decision processes (MDPs) are useful for studying a wide range of optimization problems solved via dynamic programming (DP) and reinforcement learning (RL). It provides a mathematical framework for modeling decision making in situations where outcomes are partly random and partly under the control of a decision maker. In this thesis, we formulate the proposed dynamic channel selection problem into a MDP problem and both DP and RL are used to solve the problem. Therefore, in this section, the key theories of MDP and RL are summarized.

2.3.1 Fundamentals of Markov Decision Process

In general, an MDP can be characterized by four elements, namely the state space, the action space, the state transition probability and the system reward (cost), which are defined as follows [49]:

1. $\boldsymbol{\chi} = \{\chi_1, \chi_2, \ldots \chi_N\}$: the finite space with N states. At the beginning of each period, the MDP is in one of the states.

2. For each state, there is a finite set of allowable actions $A(i)$.

3. $Pr[\chi'|\chi, a]$: the transition probability from state χ to state χ' under action a. Suppose a period begins in state χ, and an action $a \in A$ is chosen. Then with probability $Pr[\chi'|\chi, a]$, the next period's state will be χ'. The next period's state only depends on the current period's state and the selected action.

4. $g(\chi, a)$: the system reward (cost) in the state χ under action a. During a period in which the state is χ and action a is chosen, an expected reward (cost) of $g(\chi, a)$ will occur.

A policy Ω: $\chi \to A$ is a mapping from the state space χ to action space A, which determines the action to take when the system state is χ. A policy is called a stationary policy if it depends on state but not time. Given policy Ω the random process of the system state will evolve as a Markov chain. The policy induces a distribution on sequences of states. An MDP is ergodic if the associated Markov chain is ergodic for every deterministic policy [50].

The goal of the MDP problem is to find a policy π that maximize (minimize) the expected total reward (cost) over an infinite horizon, (the number of time period,) which is an infinite value. To compare policies of infinite value, two approaches are commonly used to resolve the problem of unbounded expected rewards over an infinite horizon [51].

1. Discount reward

We can discount the rewards (or costs) by assuming that a 1$ reward received during the next period will have the same value as a reward of β dollars. Therefore, a reward n steps away is discounted by β^n, where $0 < \beta < 1$ is the discount rate. Let M be the maximum reward that can be achieved during a single period. Then the maximum expected discounted reward that can be received over an infinite period horizon is:

$$M + M\beta + M\beta^2 + \ldots = \frac{M}{1-\beta} < \infty \tag{2.1}$$

Define a value function $V_\Omega(\chi)$ as the expected discounted reward earned during an infinite number of periods. Given that at the beginning of period 1, the state is χ and stationary policy is Ω. Then

$$V_\Omega(\chi) = E_\Omega(\sum_{t=1}^{t=\infty} \beta^{t-1} g(\chi(t), \Omega(\chi)) | \boldsymbol{\chi}(1) = \chi) \tag{2.2}$$

Bellman Equations for the discounted objective function is:

$$V_{\Omega}(\chi) = g(\chi, \Omega(\chi)) + \sum_{\chi' \in \boldsymbol{\chi}} Pr(\chi'|\chi, \Omega(\chi))\beta V_{\Omega}(\chi') \tag{2.3}$$

The optimal policy $\Omega^*$is obtained by solving

$$V^*_{\Omega}(\chi) = \max_{\Omega}[g(\chi, \Omega(\chi)) + \sum_{\chi' \in \boldsymbol{\chi}} Pr(\chi'|\chi, \Omega(\chi))\beta V_{\Omega}(\chi')] \tag{2.4}$$

for every state in $\boldsymbol{\chi}$.

2. Average Reward

Another way to solve the infinite horizon MDP problem is to find the policy that can maximize (minimize) the expected cost (reward) incurred per period. The average reward ρ associated with a particular policy π at a state χ is defined as:

$$\rho_{\Omega}(\chi) = \lim_{T \to \infty} \frac{1}{T} E_{\Omega}(\sum_{t=1}^{T} g_t(\chi, \Omega(\chi))) \tag{2.5}$$

For an ergodic MDP,

$$\rho_{\Omega}(\chi) = \sum_{\chi' \in \boldsymbol{\chi}} g(\chi', \Omega(\chi'))\Pi_{\Omega}(\chi') \tag{2.6}$$

where $\Pi_{\Omega}(\chi')$ is the steady-state probability of being in state χ' given policy Ω.

The Bellman equation is described as follows [52]: For any MDP that is either unichain or communicating, there exists a value function V^* and a scalar ρ^* satisfying the equation

$$V^*(\chi) + \rho^* = \max_{\Omega}[g(\chi, \Omega(\chi)) + \sum_{\chi' \in \boldsymbol{\chi}} Pr(\chi'|\chi, \Omega(\chi))V_{\Omega}(\chi')] \tag{2.7}$$

where ρ^* is the optimal average reward per period and the corresponding Ω^* is the optimal policy.

2.3.2 Solving MDPs

In this section, we will have an overview of classical solution methods for MDPs known as dynamic programming (DP).

A. Policy Itertaion

The first algorithm is called policy iteration, introduced by Howard [53]. It starts with a policy and iteratively improves it. Two steps are included: policy evaluation and policy improvement. Set an arbitrary policy Ω_0 as an initial policy and set $i = 0$.

1. Policy evaluation: For discount MDP, given a policy Ω_i, solve a set of linear equations (2.3) for each state and obtain the values $\boldsymbol{V}_{\Omega_i}$. The equations can be solved by linear equation solution methods or solved iteratively. For average reward MDP, given a policy Ω_i, solve equations (2.7) for the average reward ρ_{Ω_i} and values $\boldsymbol{V}_{\Omega_i}$ by setting the value of a reference state $V(\chi) = 0$.

2. Policy Improvement: With the obtained value function $\boldsymbol{V}_{\Omega_i}$, obtain the improved policy Ω_{i+1} by solving equation (2.4) in discounting MDP or (2.7) in average reward MDP at each state.

3. Stop if there is no change in the policy, i,e,. $\Omega_{i+1} = \Omega_i$. Otherwise increment i and go to step 1.

Consider discounted MDP as an example. The solution procedure can be describe as:

1. Initialize policy Ω with arbitrary value

2. Repeat

3. Policy evaluation: solve the linear system and obtain $V_\Omega(\chi)$
   $V_\Omega(\chi) = g(\chi, \Omega(\chi)) + \sum_{\chi' \in \boldsymbol{\chi}} Pr(\chi'|\chi, \Omega(\chi))\beta V_\Omega(\chi')\ \forall \chi \in \boldsymbol{\chi}$

4. Policy improvement: for each state $\chi \in \boldsymbol{\chi}$:
   $\Omega(\chi) \leftarrow \arg\max_a [g(\chi, \Omega(\chi)) + \sum_{\chi' \in \boldsymbol{\chi}} Pr(\chi'|\chi, \Omega(\chi))\beta V_\Omega(\chi')]$.

5. Until Ω is unchanged.

B. Value Iteration

The difficulty with policy iteration is that it requires solving N equations at every iteration, which is computationally intractable when N is large. A more attractive approach is value iteration, where the value functions are iteratively obtained until it converges. With an arbitrary function V_0 as the initial value function, the value iteration contains the following steps:

1. For each state $\chi \in \boldsymbol{\chi}$:
   $v \leftarrow V(\chi)$
   $V(\chi) \leftarrow \max_a[g(\chi, a) + \sum_{\chi' \in \boldsymbol{\chi}} Pr(\chi'|\chi, a)\beta V(\chi')]$
2. $\Delta \leftarrow \max(\Delta, |v - V(\chi)|)$
3. Stop if $\Delta < \theta$, otherwise go back to step 1.

Output a deterministic policy Ω, such that

$$\Omega(\chi) = \arg\max_a[g(\chi, a) + \sum_{\chi' \in \boldsymbol{\chi}} Pr(\chi'|\chi, a)\beta V(\chi')] \tag{2.8}$$

For average reward MDP, define $T(V)(x)$ as the right hand side of the equation (2.7), i.e.,

$$T(V)(x) = \max_a[g(\chi, a) + \sum_{\chi' \in \boldsymbol{\chi}} Pr(\chi'|\chi, a)V(\chi')] \tag{2.9}$$

The value iteration algorithm for average reward MDP is as follows:

1. For each state $\chi \in \boldsymbol{\chi}$:
   $t \leftarrow T(V)(\chi)$
   $T(V)(\chi) \leftarrow \max_a[g(\chi, a) + \sum_{\chi' \in \boldsymbol{\chi}} Pr(\chi'|\chi, a)\beta V(\chi')]$
2. $\Delta \leftarrow \max(\Delta, |t - T(V)(\chi)|)$
3. Stop if $\Delta < \theta$, otherwise go back to step 1.

The value iteration algorithm does not explicitly compute the average reward, but it can be estimated as $V^{n+1}(\chi) - V^n(\chi)$ for large n.

C. Relative Value Iteration

For average award MDP, value iteration has the disadvantage that the values $V(\chi)$ can be very large since the iterate can become unbounded, which causes numerical instability. To avoid this problem, a relative value iteration algorithm is generally used.

The Bellman equation for average reward can be written as

$$V(\chi) = \max_a [g(\chi, a) + \sum_{\chi' \in \boldsymbol{\chi}} Pr(\chi'|\chi, a)V(\chi')] - \rho \tag{2.10}$$

In the value iteration, the main transformation is obtained by setting the optimal average reward ρ to 0. In the regular version of relative value iteration, ρ is arbitrarily set to one of the current value, i.e., $\rho = V_i(\chi)$, where i denotes the current time period and χ is an arbitrarily selected state. In [54], the maximum value of the current iteration is selected as the value of ρ, i.e., $\rho = \max_{\chi \in \boldsymbol{\chi}}[V_i(\chi)]$. Then the step-by-step details of the algorithm will be:

1. Set $k = 0$ and select an arbitrary vector $V(0)$.

2. For each state $\chi \in \boldsymbol{\chi}$:
   $v \leftarrow V(\chi)$
   $V(\chi) \leftarrow \max_a [g(\chi, a) + \sum_{\chi' \in \boldsymbol{\chi}} Pr(\chi'|\chi, a)\beta V(\chi')] - \max_{\chi \in \boldsymbol{\chi}}[V(\chi)]$

3. $\Delta \leftarrow \max(\Delta, |v - V(\chi)|)$

4. Stop if $\Delta < \varepsilon$ otherwise go to step 2.

The ε-optimal policy Ω is determined by:

$$\Omega(\chi) = \arg\max_a [g(\chi, a) + \sum_{\chi' \in \boldsymbol{\chi}} Pr(\chi'|\chi, a)V(\chi')] \tag{2.11}$$

2.3.3 Reinforcement Learning

All the algorithms discussed in the last section are *model-based* which require complete knowledge of the state transition matrices as well as the expected reward or cost of each action and state pair. However, the RL algorithm is *model-free* which eliminates this requirement, and can adaptively perform decisions by learning from the environment and previous decisions.

RL is an area of machine learning inspired by behaviourist which focuses on how to take actions for the software agents to obtain the expected maximum (minimum) cumulative reward (cost) over a long run. In RL, an agent is placed in an unknown environment. By learning from its history of interaction with the environment, the agent tries to learn the optimal policy. There exists a variety of RL algorithms. Among them Q-learning is the best studied discounted RL method and has been widely used in solving MDP problems. It was first introduced by Watkins in 1989 [55]. The first average-reward RL method, R-learning, was proposed by Schwartz [56]. It has been proved that it outperforms Q-learning if the parameters are well adjusted [57]. In this section, a brief overview of the two main RL method are presented.

A. Q-learning

A history of an agent is a sequence of state-action-reward which can be represented by a tuple $< \chi, a, r, \chi' >$. It means that the agent was in state χ, took action a and received reward r. As a result, it went into state χ'.

Define $Q^*(\chi, a)$ as the expected value (cumulative discount reward) of taking action a in state χ and then following the optimal policy. In Q-learning, the agent maintains a table of $Q[\boldsymbol{\chi}, A]$, where $Q[\chi, a]$ represents its current estimate of $Q^*(\chi, a)$. The value of $Q^*(\chi, a)$ is estimated according to

$$Q[\chi, a] \leftarrow Q[\chi, a] + \alpha(r + \gamma \max_{a'} Q[\chi', a'] - Q[\chi, a]) \tag{2.12}$$

The procedure of the algorithm is:

1. Initialize $Q(\chi, a)$ arbitrarily

2. Repeat (for each episode)

   Initialize χ

3. Repeat (for each step of episode)

   Selection action a according to certain exploration policy and observe reward r and state χ'.

   Update $Q[\chi, a]$ with equation (2.12)

   $\chi \leftarrow \chi'$

4. Until termination

The parameters used in the update process are defined as:

- α denotes the learning rate which is set between 0 and 1. 0 means no learning since the Q-values are never updated. The higher value it is set the quicker the learning occurs.

- γ is the discount factor which is also between 0 and 1. It models the fact that the future rewards worth less than the immediate rewards. Convergence requires the discount factor to be less than 1.

The convergence proof of Q-learning algorithm was presented by Watkins and Dayan in 1992 [55]. The algorithm learns an optimal policy no matter which policy it is following, as long as every action and state pair has been tried unlimited times. However, it may suffer from slow rate of convergence especially when the discount factor γ is close to one.

B. R-Learning for Undiscounted Continuing Tasks

R-learning Similar to the definition of Q-learning, the value function $R^{\Omega}(\chi, a)$ represents the average value of taking action a at state χ and then following the policy Ω. $R(\chi, a)$ is defined as [57]:

$$R^{\Omega}(\chi, a) \leftarrow r(\chi, a) - \rho^{\Omega} + \sum_{\chi'} P(\chi'|\chi, a)V^{\Omega}(\chi') \tag{2.13}$$

where $V^{\Omega}(\chi') = \max_{a \in A} R^{\Omega}(\chi', a)$ and $r(\chi, a)$ is the average reward of policy Ω. R-learning algorithm consists of the following steps:

1. Initialize $R(\chi, a)$ with an arbitrary value

2. Repeat (for each episode)
   Initialize state χ

3. Repeat (for each step of episode)
   Choose action $a = \arg\max R(\chi, a)$ with some probability. Otherwise choose a

random exploratory action. Observe the reward r and the next state χ'.

Update the R values with the following rules:

$$R(\chi, a) \leftarrow R(\chi, a)(1 - \beta) + \beta(r - \rho + \max_a R(\chi', a)) \tag{2.14}$$

If a non-exploratory action is performed, the average reward ρ is updated according to:

$$\rho \leftarrow \rho(1 - \alpha) + \alpha[r + \max_a R(\chi', a) - \max_a R(\chi, a)] \tag{2.15}$$

4. Until termination

Here, β is the learning rate for the action values between 0 and 1 and α is the learning rate for updating ρ which is also between 0 and 1.

C. Exploration Strategies

To try all the actions in every state to achieve convergence, RL methods apply exploration strategy to occasionally take sub-optimal actions. Exploration methods can be divided into undirected and directed methods. Undirected methods select a random action without considering the results from learning. Directed methods decide which states to explore according to the results of learning. A detailed comparison of these two methods was given in [57]. We will briefly introduce one undirect exploration method (semi-uniform exploration) and one directed exploration method (uncertainty exploration (UE)) as examples.

- *Semi-Uniform Exploration*: Let $U(\chi, a)$ denote a generic value function which could be $Q(\chi, a)$ or $R(\chi, a)$. The best action a that maximizes the value function $U(\chi, a)$ is selected with a fixed probability p_{exp}. With probability $1 - p_{exp}$, a random action is applied.

- *UE exploration*: The agent selects action $a = \arg\max(\chi, a) + \frac{c}{N_f(\chi,a)}$ with probability p and picks a random action with probability $1 - p$.

The configuration of learning rate parameters α and β is also important when applying the algorithms. A detailed sensitivity analysis of R-learning was conducted in paper [57]. In the paper, the authors compared the two algorithm and two findings were discovered: *R-learning is more sensitive than Q-learning to exploration strategies and can get trapped in limit cycles; however, R-learning can be fine-tuned to outperform Q-learning in two domains where the comparison experiments were carried out*.

Chapter 3

Dynamic Load-balancing Spectrum Decision for Multi-channel Cognitive Radio Networks with Heterogeneous Services

In this chapter, we study dynamic load-balancing spectrum decision for cognitive radio networks that dynamically distributes packets from an SU to different available primary channels. We consider two different classes of services at the SU, i.e., delay sensitive (DS) and best effort (BE) services, and assign a higher priority to the DS services. We propose a new queueing analytical model to address this priority issue and analyze delay performance for the two services separately. Based on the analytical results, two Markov decision processes (MDPs) are formulated with objectives to minimize the average delay of both services while guaranteeing the priority of the DS services. Reinforcement learning (RL) is applied to find the optimal solutions when the traffic and channel characteristics are unknown. To address the computational complexity issue in the MDP solutions, we proposed a myopic scheme based on the estimated packet sojourn time, which can be calculated by formulating a phase type distribution. Simulation results show that the

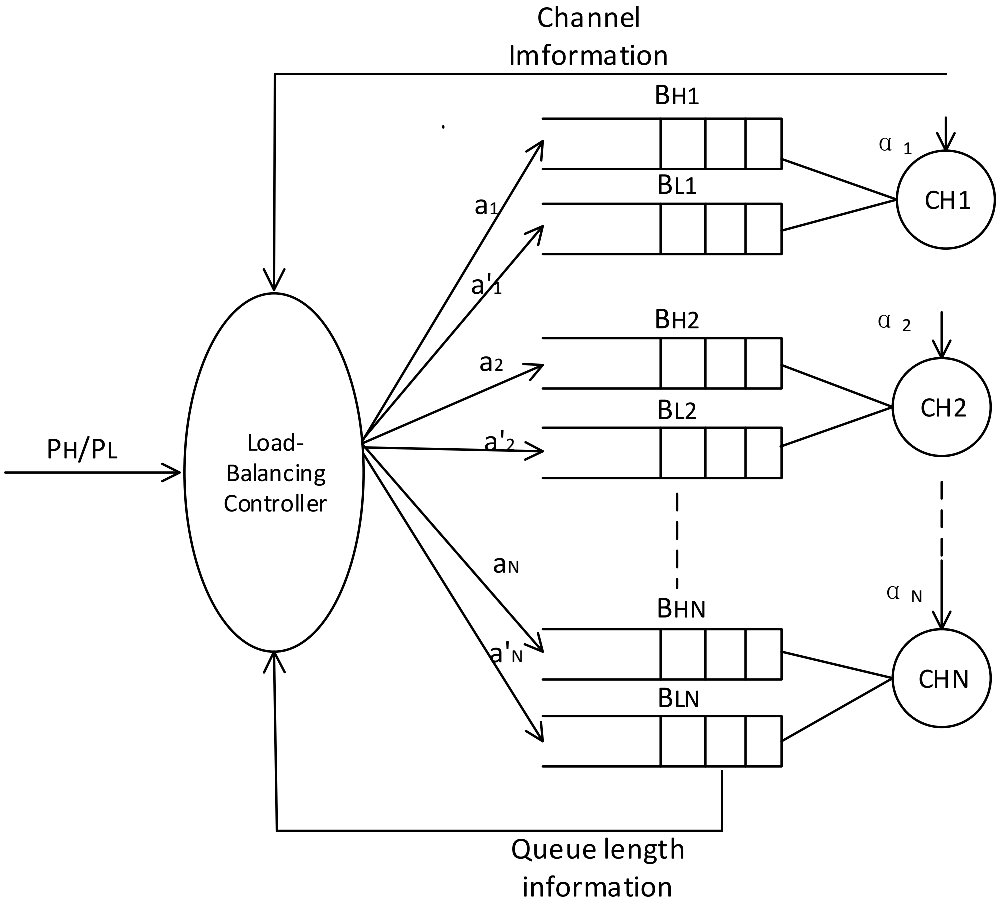

Figure 3.1: Load-balancing model

proposed myopic scheme can achieve significant reduction on computational complexity with a cost on the delay performance of low priority BE services.

3.1 System Model

We consider a time-slotted CR system, which consists of N independent PUs and one SU. Each channel is allocated to one PU. We assume that the SU is equipped with multiple receiving and transmitting antennas and can access all the N channels. Both BE and DS services are supported by the SU. At the beginning of each time slot, spectrum sensing is performed for all the N channels and the SU's packets will be transmitted only if the channel is sensed idle. Perfect sensing is assumed in the paper [8] [48]. The duration of each time slot is denoted by Δt and the required sensing time is τ. The arrival processes of SU's DS and BE packets are assumed to follow independent and identically distributed (i.i.d) Bernoulli processes in each time slot, with parameters p_H and p_L, respectively. Let $1/q_{Hn}$ and $1/q_{Ln}$ denote the average service times of the SU's DS and BE packets at channel n, respectively, which are assumed to follow geometric distributions [43]. We first assume that the arrival probabilities and service times of the PUs are known to the SUs based on the collected empirical data. The situations that PUs' statistical information is unknown will be discussed in section 3.4. For the explanation purpose, we assume that the arrival processes of the PU's packets at channel n follows Bernoulli distribution with rate α_n, and the service times follow geometric distribution with parameter $1/\beta_n$. We consider block fading and the channel fading coefficient of channel n, h_n, follows an i.i.d distribution among time slots but remains quasi-static within each time slot. Then, the maximum achievable data rate of the SU at channel n is

$$R_n = \log(1 + \gamma_n |h_n|^2), \tag{3.1}$$

where γ_n is the received signal-to-noise ratio (SNR) when the channel gain is equal to unity. Let the packet length of both SU's services be d bits. Then, in order to guarantee that one packet can be transmitted within one time slot, the required packet transmission rate is

$$R_{req} = \frac{d}{\Delta t - \tau}. \tag{3.2}$$

Thus, q_{Hn} and q_{Ln} at channel n can be calculated as:

$$q_{Hn} = q_{Ln} = Pr\{R_{req} < R_n\}. \tag{3.3}$$

A. Queueing Model and Dynamic Channel Access Scheme

To evaluate the effect of prioritize channel access, traffic rate and channel conditions, we propose the following priority queue model. We consider that the SU maintains one finite buffer for each channel to buffer both interrupted packets and the packets that can not be immediately served. As illustrated in Fig.3.1, for each channel, each secondary buffer consists of two virtual queues. One is high priority queue B_{Hn} for the DS packets and the other is low priority queue B_{Ln} for the BE packets. The buffers' lengths for the DS and BE services are K_H and K_L, respectively. We assume each PU has one buffer at the assigned channel with length K_p to capture its potential dynamics. All the queues in the system adopt the first come first service (FCFS) protocol.

It is worth mentioning that the proposed system model can be applied to the scenario where the SU is a broadband user who opportunistically accesses N narrow-band primary links. In such case, the SU transmits data via orthogonal frequency-division multiple access (OFDMA) scheme. Moreover, the system model can also be adapted to the cases where a secondary network consists of multiple SUs with homogeneous arrival and service distributions [43].

Upon each packet's arrival at the beginning of a time slot, a decision has to be made on which channel to be selected for transmission. After the decision, the packet will be delivered to the corresponding buffer and waits for the chance to access to the channel. During the transmission of SU's packet, if the PU appears on the channel, the SU has to stop its transmission immediately and retransmit the packet until the channel becomes idle again. The BE packets transmission will be interrupted if a DS packet tries to access the same channel. The interrupted packet will be resumed and re-transmitted when the channel is idle again and no DS packets waiting in the queue. Therefore, the packets in the low priority queue can not be transmitted unless the high priority queue is empty.

We further define the following parameters.

i) Let a_n and $a'_n \in \{0, 1\}$ indicate whether channel n is selected at the current slot for the DS and BE packets, respectively. $a_n = 1$ ($a'_n = 1$) denotes channel n is selected for the DS (BE) packet. Otherwise, $a_n = 0$ ($a'_n = 0$).

ii) Let $\boldsymbol{Q}_H = \{Q_{H1}(t), \ldots, Q_{HN}(t)\}$ and $\boldsymbol{Q}_L = \{Q_{L1}(t), \ldots, Q_{LN}(t)\}$ be the secondary queue lengths over all N channels for the DS and BE services, respectively, where $Q_{Hn}(t)$ and $Q_{Ln}(t)$ denote the unfinished number of packets in queue B_{Hn} and B_{Ln}, respectively, at the beginning of the t-th time slot;

iii) Let $\boldsymbol{S}_t = \{S_1(t), \ldots, S_N(t)\}$ be the set of states of all the N channels, where $S_n(t) = 0$ denotes that the channel is idle, i.e, the PU is not occupying the channel; otherwise, $S_n(t) = 1$.

At the beginning of each slot, the SU dynamically makes channel selection decisions based on the channel state and the joint queue length states. The decision policies will be derived in the following sections.

3.2 Queueing Analysis

To jointly consider the PU's activity, channel capacity and buffer states, in this section, we develop queueing models to describe the behaviors of the PU activity, and both the SU's DS and BE services.

A. Primary User Activity

To capture all the channels' ON/OFF behavior according to the PUs' occupancy, we build a Markov chain for each PU based on their arrival and service processes. Let $N_n(t)$ denote the number of packets in the queue of PU n. Then, $N_n(t)$ follows a Markov chain with state space $\{k_p, 0 \leq k_p \leq K_p + 1\}$, and the transition matrix of $N_n(t)$ is:

$$\mathbf{P}_n = \begin{bmatrix} \bar{\alpha}_n & \alpha_n & & \\ \bar{\alpha}_n\beta_n & \bar{\alpha}_n\bar{\beta}_n + \alpha_n\beta_n & \alpha_n\bar{\beta}_n & \\ & \ddots & \ddots & \ddots \\ & & \bar{\alpha}_n\beta_n & \bar{\alpha}_n\bar{\beta}_n + \alpha_n \end{bmatrix}, \tag{3.4}$$

where $\bar{\alpha}_n = 1 - \alpha_n$ and $\bar{\beta} = 1 - \beta_n$. The derivation process considers following cases.

- The state of $N_n(t)$ stays zero when the queue length of channel n is zero at time k and remains zero at time $k+1$, which means that no packet arrives at the beginning of slot $k+1$. The probability that no packet arrives is $\bar{\alpha}_n$.
- The state of $N_n(t)$ transitions from 0 to 1, if one packet arrives. The probability is α_n.

- The state of $N_n(t)$ transitions from 1 to 0, if no packet arrives and one packet leaves. The probability is $\bar{\alpha}_n \beta_n$. The same probability applies when the state of $N_N(t)$ transitions from any $n \geq 1$ to $n-1$.

- If the state of $N_n(t)$ stays at $n \neq 0$, it means that no packet arrives and no packet leaves with a probability of $\bar{\alpha}_n \bar{\beta}_n$, or one packet arrives and one packet leaves with a probability of $\alpha_n \beta_n$. Thus, the overall transition probability is $\bar{\alpha}_n \bar{\beta}_n + \alpha_n \beta_n$.

- The state of $N_n(t)$ transitions from $n \geq 1$ to $n+1$, if one packet arrives and no packet leaves. The probability is $\alpha_n \bar{\beta}_n$.

- If the state of $N_n(t)$ stays at K , it means that one packet arrives and no packet leaves, or one packet arrives but the buffer is full. The overall probability is $\bar{\alpha}_n \bar{\beta}_n + \alpha_n$.

Let $x_i^{(k)}$ be the probability that there are i packets in the queue of PU n at time slot k, and $\boldsymbol{x}^{(k)} = [x_0^{(k)}, x_1^{(k)}, \ldots]$. Then we have

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} \mathbf{P_n} \text{ or } \boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(0)} \mathbf{P_n}^{k+1}. \tag{3.5}$$

Given that each PU's queue is stable, there exists a steady state distribution $\boldsymbol{x} = [\pi_0, \pi_1, \ldots, \pi_K]$ such that

$$\boldsymbol{x}\mathbf{P_n} = \boldsymbol{x},\ \boldsymbol{x}\mathbf{1} = 1. \tag{3.6}$$

The vector $\boldsymbol{x}$ can be solved by applying the Matrix-analytic approach [58].

Given the transition matrix of $N_n(t), n = 1, 2, \ldots, N$, we can derive the transition matrix for each PU channel's ON/OFF states, where OFF denotes that the channel is idle, i.e., the number of packets in the PU buffer is 0, and ON otherwise. We can rewrite the transition matrix (3.4) in the following format:

$$\mathbf{P_n} = \begin{bmatrix} \bar{\alpha}_n & \alpha_n \boldsymbol{\phi} \\ \boldsymbol{v} & \boldsymbol{V} \end{bmatrix},$$

where $\boldsymbol{\phi} = [1, 0 \ldots, 0]$ with a dimension of K_p, $\boldsymbol{V}$ is a $K_p \times K_p$ matrix as:

$$\boldsymbol{V} = \begin{bmatrix} \bar{\alpha}_n\bar{\beta}_n + \alpha_n\beta_n & \alpha_n\bar{\beta}_n & & \\ \bar{\alpha}_n\beta_n & \bar{\alpha}_n\bar{\beta}_n + \alpha_n\beta_n & \alpha_n\bar{\beta}_n & \\ & \ddots & \ddots & \ddots \\ & & \bar{\alpha}_n\beta_n & \bar{\alpha}_n\bar{\beta}_n + \alpha_n \end{bmatrix},$$

$\boldsymbol{v} = \mathbf{1} - \boldsymbol{V}\mathbf{1}$, where $\mathbf{1}$ is a K-dimensional column vector. Let $\boldsymbol{v} = [v_1, \ldots, v_K]$ and $\boldsymbol{V} = [\boldsymbol{V}_1, \ldots, \boldsymbol{V}_K]$. We use 0 to denote the OFF state and 1 for the ON state. Then the transition probabilities of channel n between ON/OFF states can be calculated as:

$$P_{00} = \pi_0\bar{\alpha}_n, \;\; P_{01} = \pi_0\alpha_n\boldsymbol{\phi}\boldsymbol{e},$$

$$P_{10} = \frac{\sum_{i=1}^{K}\pi_i v_i}{\sum_{i=1}^{K}\pi_i}, \;\; P_{11} = \frac{\sum_{i=1}^{K}\pi_i \boldsymbol{V}\boldsymbol{e}}{\sum_{i=1}^{K}\pi_i}.$$

B. Queueing Analysis of the DS Service

To make channel selection decision for the DS packets, we need to consider the PU's activity, i.e., each channel's ON/OFF state, each channel's service rate, and the number of packets in each high priority queue. With this consideration, we develop a queueing model that combine all these three factors. Since the DS packets have a higher priority than the BE packets, the BE packets in the low priority queue have no influence on the transmission of the DS packets.

Define $\boldsymbol{\chi}(t) = (\chi_1(t), \ldots, \chi_N(t))$ as the overall system state for the DS packets at the t-th slot, where $\chi_n(t) = (Q_{Hn}(t), S_n(t))$ denotes the system state of channel n. Let $a_H(n)$ and $b_H(n)$ be the DS packets' arrival probability and departure probability, respectively, in each time slot at channel n. Define k_{hn} and s_n as the number of packets in queue B_{Hn} and the state of channel n, respectively. Then it is easy to prove that $\chi_n(t)$ is a discrete time Markov chain (DTMC), and its state space is $\Delta = \{(k_{hn}, s_n) | 0 \leq k_{hn} \leq K_H + 1, s_n =$

$0\, or\, 1\}$. Therefore, the transition matrix for $\chi_n(t)$ can be derived as:

$$\mathbf{P_{\chi_n}} = \begin{bmatrix} B_1 & B_0 & & & & \\ A_2 & A_1 & A_0 & & & \\ & A_2 & A_1 & A_0 & & \\ & & \ddots & \ddots & & \ddots \\ & & & A_2 & A_1 + A_0 & \end{bmatrix}, \tag{3.7}$$

where

$$\mathbf{B_1} = \bar{p_H} \begin{bmatrix} P_{00} & P_{01} \\ P_{10} & P_{11} \end{bmatrix}, \ \mathbf{B_0} = p_H \begin{bmatrix} P_{00} & P_{01} \\ P_{10} & P_{11} \end{bmatrix},$$

$$\mathbf{A_2} = \begin{bmatrix} \bar{a}_s(n) b_s(n) P_{00} & \bar{a}_s(n) b_s(n) P_{01} \\ 0 & 0 \end{bmatrix},$$

$$\mathbf{A_0} = \begin{bmatrix} a_s(n) \bar{b}_s(n) P_{00} & a_s(n) \bar{b}_s(n) P_{01} \\ a_s(n) P_{10} & a_s(n) P_{11} \end{bmatrix},$$

$$\mathbf{A_1} = \begin{bmatrix} P_{(k_{hn},0)\to(k_{hn},0)} & P_{(k_{hn},0)\to(k_{hn},1)} \\ P_{(k_{hn},1)\to(k_{hn},0)} & P_{(k_{hn},1)\to(k_{hn},1)} \end{bmatrix},$$

with

$$P_{(k_{hn},0)\to(k_{hn},0)} = (\bar{a}_s(n)\bar{b}_s(n) + a_s(n) b_s(n)) P_{00},$$
$$P_{(k_{hn},0)\to(k_{hn},1)} = (\bar{a}_s(n)\bar{b}_s(n) + a_s(n) b_s(n)) P_{01},$$
$$P_{(k_{hn},1)\to(k_{hn},0)} = \bar{a}_s(n) P_{10}, P_{(k_{hn},1)\to(k_{hn},1)} = \bar{a}_s(n) P_{11}.$$

The submatrices in (3.7) are explained as follows:

- B_1 is a block that denotes the probability of $Q_{Hn}(t)$ remaining at zero at time k and $k+1$, while the channel state is changing between ON and OFF. Since the arrival process is independent from the channel states, B_1 can be represented by the product of the probability of no arrival in one slot $\bar{p_H}$ and the transition matrix of the channel state.

- Block B_0 denotes the probability that the state of $Q_{Hn}(t)$ increases by 1 from zero in two consecutive time slots. This event means that there is one arriving DS packet.

- Block A_2 denotes the transition of $Q_{Hn}(t)$ from state k_{hn} to state $k_{hn} - 1$ while $k_{kn} > 0$. Thus, if the channel is OFF, the transition happens when one packet completes service and no new packet arrives. If the channel state is ON, since no packets can be transmitted, the probability of having one less packet in the queue is 0.

- Block A_1 denotes that the state of $Q_{Hn}(t)$ remains unchanged from one slot to the next one when $k_{kn} > 0$. When the channel is OFF, the transition can happen when no packet leaves and no packet arrives, or one packet arrives and one packet leaves. When the channel is ON, it can happen only when no packet arrives.

- Block A_0 denotes that the state of $Q_{Hn}(t)$ increases by one when $k_{kn} > 0$. The transmission becomes feasible only when one packet arrives while no packet leaves or there is one packet arrived.

C. Queueing Analysis of the BE Service

The delay of the BE packets in the low priority queue is influenced by the BE packets that are already in the queue, the DS packets that are already in the high priority queue, the newly arriving DS packets during waiting time, the channel states and the service rates. In order to study the queueing dynamics of the BE service, we propose a Markov analysis model which combines the high priority queue for DS packets and the low priority queue for BE packets with the ON/OFF server.

Define a Markov chain $\boldsymbol{\psi}(t) = (\psi_1(t), \ldots, \psi_N(t))$ as the overall system state for the BE packets at the t-th slot, where $\psi_n(t) = (Q_{Ln}(t), Q_{Hn}(t), S_n(t))$ is the system state of channel n. Let k_{ln} and k_{hn} denote the number of packets in the high and low priority queues, respectively, and s_n denote the channel state. The state space of this Markov chain becomes $\Delta = \{(k_{ln}, k_{hn}, s_n) | 0 \leq k_{ln} \leq K_L + 1, 0 \leq k_{hn} \leq K_H + 1, s_n = 0 \, or \, 1\}$. At each time slot, the SU selects a target channel for the arriving BE packets after observing the global system state $\boldsymbol{\psi}(t)$. Let $a_L(n)$ and $b_L(n)$ denote the arrival and departure probabilities of the BE packets at each time slot for channel n, respectively. Then, the transition matrix for $\boldsymbol{\psi}_n(t)$ can be formulated as:

$$\mathbf{P}_{\boldsymbol{\psi}_\mathbf{n}} = \begin{bmatrix} B_1' & B_0' & & & \\ A_2' & A_1' & A_0' & & \\ & A_2' & A_1' & A_0' & \\ & & \ddots & \ddots & \ddots \\ & & & A_2' & A_1' + A_0' \end{bmatrix}, \tag{3.8}$$

where $\mathbf{B_1'} = \bar{a}_L \mathbf{P}_{\boldsymbol{\chi}_\mathbf{n}}, \mathbf{B_0'} = a_L \mathbf{P}_{\boldsymbol{\chi}_\mathbf{n}}$, $a_s(n) = a_H(n)$, $b_s(n) = b_H(n)$,

$$\mathbf{A_2'} = \begin{bmatrix} B_2^1 & B_2^0 & & \\ & 0 & 0 & \\ & & \ddots & \ddots, \end{bmatrix}, \tag{3.9}$$

$$\mathbf{A_1'} = \begin{bmatrix} B_1^1 & B_1^0 & & \\ A_1^2 & A_1^1 & A_1^0 & \\ & \ddots & \ddots & \ddots \\ & & A_1^2 & A_1^1 + A_1^0 \end{bmatrix}, \mathbf{A_0'} = \begin{bmatrix} B_0^1 & B_0^0 & & \\ A_0^2 & A_0^1 & A_0^0 & \\ & \ddots & \ddots & \ddots \\ & & A_0^2 & A_0^1 + A_0^0 \end{bmatrix},$$

$$\mathbf{B_2^1} = \begin{bmatrix} \bar{a}_L(n) b_L(n) \bar{a}_H(n) P_{00} & \bar{a}_L(n) b_L(n) \bar{a}_H(n)) P_{01} \\ 0 & 0 \end{bmatrix},$$

$$\mathbf{B_2^0} = \begin{bmatrix} \bar{a}_L(n) b_L(n) a_H(n) P_{00} & \bar{a}_L(n) b_L(n) a_H(n) P_{01} \\ 0 & 0 \end{bmatrix},$$

$$\mathbf{B_0^1} = \begin{bmatrix} a_L(n)\bar{b}_L(n)\bar{a}_H(n)P_{00} & a_L(n)\bar{b}_L(n)\bar{a}_H(n)P_{01} \\ a_L(n)\bar{a}_H(n)P_{10} & a_L(n)\bar{a}_H(n)P_{11} \end{bmatrix},$$

$$\mathbf{B_0^0} = \begin{bmatrix} a_L(n)\bar{b}_L(n)a_H(n)P_{00} & a_L(n)\bar{b}_L(n)a_H(n)P_{01} \\ a_L(n)a_H(n)P_{10} & a_L(n)a_H(n)P_{11} \end{bmatrix},$$

$$\mathbf{B_1^1} = \begin{bmatrix} B_{10}^{10} & B_{10}^{11} \\ B_{11}^{10} & B_{11}^{11} \end{bmatrix}, \quad \mathbf{B_1^0} = \begin{bmatrix} B_{10}^{00} & B_{10}^{01} \\ B_{11}^{00} & B_{11}^{01} \end{bmatrix},$$

with

$$B_{10}^{10} = (\bar{a}_L(n)\bar{b}_L(n) + a_L(n)b_L(n))\bar{a}_H(n)P_{00},$$

$$B_{10}^{11} = (\bar{a}_L(n)\bar{b}_L(n) + a_L(n)b_L(n))\bar{a}_H(n)P_{01},$$

$$B_{11}^{10} = \bar{a}_L(n)\bar{a}_H(n)P_{10}, B_{11}^{11} = \bar{a}_L(n)\bar{a}_H(n)P_{11},$$

$$B_{10}^{00} = (\bar{a}_L(n)\bar{b}_L(n) + a_L(n)b_L(n))a_H(n)P_{00},$$

$$B_{10}^{01} = (\bar{a}_L(n)\bar{b}_L(n) + a_L(n)b_L(n))a_H(n)P_{01},$$

$$B_{11}^{00} = \bar{a}_L(n)a_H(n)P_{10}, B_{11}^{01} = \bar{a}_L(n)a_H(n)P_{11},$$

$$\boldsymbol{A_1^2} = \bar{a}_L(n)\boldsymbol{A_2},\ \boldsymbol{A_1^1} = \bar{a}_L(n)\boldsymbol{A_1},\ \boldsymbol{A_1^0} = \bar{a}_L(n)\boldsymbol{A_0},$$

$$\boldsymbol{A_0^2} = a_L(n)\boldsymbol{A_2},\ \boldsymbol{A_0^1} = a_L(n)\boldsymbol{A_1},\ \boldsymbol{A_0^0} = a_L(n)\boldsymbol{A_0}.$$

We use A_0' as an example to show the derivation processes of all submatrices.

Block A_0' denotes the transition matrix of Q_{Ln} from state k_{ln} to state $k_{ln}+1$ when $k_{ln} > 0$, which further consists of several sub-blocks. The sub-block B_0^1 denotes the probability of Q_{Hn} staying at 0. For the case that the channel state is OFF, one BE packet will receive service since no DS packets is waiting, and the system state will transition from $(k_{ln}, 0)$ to $(k_{ln} + 1, 0)$, $k_{ln} > 0$, under the conditions that the BE packet that is under service has not been completed, a new BE packet arrives, and no DS packet arrives. For the case that the channel state is ON, no packets can be served. Hence, the system state transitions from

$(k_{ln}, 0)$ to $(k_{ln} + 1, 0)$, $k_{ln} > 0$, if one BE packet arrives and there is no newly arrived DS packet. When the channel state is ON, no BE packets can leave. Therefore, the state of Q_{Ln} increases by one only if one BE packet arrives. The development of sub-block B_0^0 is in a similar way, except that the state of Q_{Hn} increases by 1 due to one DS packet's arrival.

When Q_{Hn} is larger than zero, i.e., there are packets in the high priority queue, the BE packets can not be served. Therefore, only if one BE packet arrives, the state of Q_{Ln} will change from k_{ln} to $k_{ln} + 1$. Since the arrival process of BE packets is independent of the service and arrival processes of the DS packets, sub-block A_0^2, A_0^1 and A_0^2 can be represented by the product of the BE packets' arrival probability and the transition matrices A_0, A_1, and A_2 of the DS packets, respectively.

D. Problem Formulation

Given observed system states $\boldsymbol{\chi}(t)$ and $\boldsymbol{\psi}(t)$, the SU makes channel selection decisions for the DS and BE packets, respectively, according to a stationary policy defined below.

Definition 1. (Stationary Channel Selection Policy) A stationary channel selection policy is a mapping from the system states to channel selection actions. The action space is $A = \{0, 1, 2, \ldots, n\}$ where $A = n, n \neq 0$ means channel n is selected, $A = 0$ means all the buffer is full and the packet is dropped. Only one channel can be selected for each packet.

Denote Ω_{DS} as a stationary policy for the DS packets. Given a feasible unichain policy Ω_{DS}, the induced Markov chain $\{\boldsymbol{\chi}(t)\}$ is ergodic so that there exists a unique steady state distribution $\pi(\Omega_{DS})$ [49]. Assume that the arrival rate falls inside the stability region of the system and we use average queue length as an approximate measurement for average delay [49]. The delay-optimal channel selection problem for the DS service is formulated as the following optimization problem.

Problem 1. (Delay-Optimal Policy for the DS Services)

$$\min_{\Omega_{DS}} \mathbb{E}^{\pi(\Omega_{DS})}[\sum_{n} Q_{Hn}]. \tag{3.10}$$

where $\mathbb{E}^{\Omega_{DS}}$ denotes the expectation operator with respect to the probability measure induced by policy Ω_{DS}.

Based on same analysis, the delay-optimal channel selection problem for the BE service can be formulated.

Problem 2. (Delay-Optimal Policy for the BE Services)

$$\min_{\Omega_{BE}} \mathbb{E}^{\pi(\Omega_{BE})}[\sum_{n} Q_{Ln}]. \tag{3.11}$$

3.3 Markov Decision Problem Formulation

In this section we shall formulate the delay minimization problems in (3.10) and (3.11) as infinite horizon Markov Decision Problems (MDPs) to obtain optimal policy channel selection policies for both SU's services.

A. MDP formulation for the DS services

A stationary control policy induces a random process $\boldsymbol{\chi}(\boldsymbol{t})$. We can show that $\boldsymbol{\chi}(\boldsymbol{t})$ is a Markov chain. Let $Pr[\boldsymbol{\chi}'|\boldsymbol{\chi}, \Omega_{DS}(\boldsymbol{\chi})]$ represent the transition probability from the current state $\boldsymbol{\chi}$ to the next state $\boldsymbol{\chi}'$ when action $\Omega_{DS}(\boldsymbol{\chi})$ is taken, where $\Omega_{DS}(\boldsymbol{\chi})$ represents the action that is taken based on policy Ω_{DS} when the system state is $\boldsymbol{\chi}$. The conditional transition probabilities $Pr[\boldsymbol{\chi}'|\boldsymbol{\chi}, \Omega_{DS}(\boldsymbol{\chi})]$ can be calculated based on the queueing dynamics derived in the previous section. Specifically, for channel n, the transition probability $Pr[\chi_n'|\chi_n, \Omega_{DS}(\boldsymbol{\chi})]$ can be derived from (3.7) with

$$a_s(n) = \begin{cases} p_H & \text{if } \Omega_{DS}(\boldsymbol{\chi}) = n \\ 0 & \text{otherwise} \end{cases},$$

$$b_s(n) = q_{Hn}.$$

It means that if channel n is selected as the transmission channel for the coming packet, the probabilities of one packet arrives at the current time slot are p_H at channel n, and 0 at any other channel.

Since $\chi_n(t)$ is independent of n once the transmission channel is selected, the joint transition probabilities $Pr[\boldsymbol{\chi}'|\boldsymbol{\chi}, \Omega_{DS}(\boldsymbol{\chi})]$ can be derived through kronecker product of the transition probability of each channel.

$$\begin{aligned} &Pr[\boldsymbol{\chi}'|\boldsymbol{\chi}, \Omega_{DS}(\boldsymbol{\chi})] \\ =&Pr[\chi_1'|\chi_1, \Omega_{DS}(\boldsymbol{\chi})] \otimes Pr[\chi_2'|\chi_2, \Omega_{DS}(\boldsymbol{\chi})] \otimes \ldots \otimes Pr[\chi_N'|\chi_N, \Omega_{DS}(\boldsymbol{\chi})]. \end{aligned} \tag{3.12}$$

Hence, given a feasible unichain policy Ω_{DS}, $\boldsymbol{\chi}$ is a ergodic Markov chain and we can rewrite the delay minimum problem for the DS service in (3.10) as a MDP problem

$$\min_{\Omega_{DS}} \mathbb{E}^{\Omega_{DS}}[g(\boldsymbol{\chi}, \Omega_{DS}(\boldsymbol{\chi}), \boldsymbol{\chi}')], \tag{3.13}$$

where $\boldsymbol{\chi}' = \{\chi_1', \ldots, \chi_N'\}$ with $\chi_n' = \{Q_{Hn}', S_n'\}$, $g(\boldsymbol{\chi}, \Omega_{DS}(\boldsymbol{\chi}), \boldsymbol{\chi}')$ is the per-stage delay cost which can be calculated as

$$g(\boldsymbol{\chi}, \Omega_{DS}(\boldsymbol{\chi}), \boldsymbol{\chi}') = \sum_n Q_{Hn}'. \tag{3.14}$$

The delay optimal policy Ω_{DS} can be obtained by solving the following Bellman equation [52]

$$\theta + V(\boldsymbol{\chi}) = \min_{\Omega_{DS}} \sum_{\boldsymbol{\chi}'} Pr[\boldsymbol{\chi}'|\boldsymbol{\chi}, \Omega_{DS}(\boldsymbol{\chi})][g(\boldsymbol{\chi}, \Omega_{DS}(\boldsymbol{\chi}), \boldsymbol{\chi}') + V(\boldsymbol{\chi}')]. \tag{3.15}$$

If there exists a pair $(\theta^*, \{V^*(\boldsymbol{\chi})\})$ satisfying (3.15), then θ^* is the optimal average cost (delay) per stage, $V^*(\boldsymbol{\chi})$ is the total expected cost at the end of the process if it starts from state $\boldsymbol{\chi}$. $(\theta^*, \{V^*(\boldsymbol{\chi})\})$ can be obtained using relative value iteration. Once $(\theta^*, \{V^*(\boldsymbol{\chi})\})$ is settled, the corresponding Ω_{DS}^* is the optimal policy which can be obtained from

$$\Omega_{DS}^* = \arg\min_{\Omega_{DS}} \sum_{\boldsymbol{\chi}'} Pr[\boldsymbol{\chi}'|\boldsymbol{\chi}, \Omega_{DS}(\boldsymbol{\chi})][g(\boldsymbol{\chi}, \Omega_{DS}(\boldsymbol{\chi}), \boldsymbol{\chi}') + V^*(\boldsymbol{\chi}')]. \tag{3.16}$$

B. MDP formulation for the BE services

Using a similar method, we formulate the channel selection problem for the BE services as the following MDP with state space $\boldsymbol{\psi}(t) = \{\boldsymbol{Q}_L(t), \boldsymbol{Q}_H(t), \boldsymbol{S}(t)\}$ and action space A.

Define channel selection probability vector $P^{\Omega_{DS}} = \{p_1, p_2, \ldots, p_N\}$ where p_n denotes the probability that channel n is selected under policy Ω_{DS}. Then the arrival process of the DS packets at channel n follows Bernoulli process with a parameter $p_{cH}(n) = p_H \times p_n$. Therefore, the state transition probability at channel n, $Pr[\psi'_n|\psi_n, \Omega_{BE}(\boldsymbol{\psi})]$, can be derived from (3.8) with

$$a_H(n) = p_{cH}(n), b_H(n) = q_H(n), b_L(n) = q_L(n),$$
$$a_L(n) = \begin{cases} p_L & \text{if } \Omega_{BE}(\boldsymbol{\psi}) = n \\ 0 & \text{otherwise} \end{cases}.$$

Since $\psi_n(t)$ is an independent process at each channel, $Pr[\boldsymbol{\psi}'|\boldsymbol{\psi}, \Omega_{BE}(\boldsymbol{\psi})]$ can be calculated as

$$\begin{aligned} &Pr[\boldsymbol{\psi}'|\boldsymbol{\psi}, \Omega_{BE}(\boldsymbol{\psi})] \\ =&Pr[\psi'_1|\psi_1, \Omega_{BE}(\boldsymbol{\psi})] \otimes Pr[\psi'_2|\psi_2, \Omega_{BE}(\boldsymbol{\psi})] \otimes \ldots \otimes Pr[\psi'_N|\psi_N, \Omega_{BE}(\boldsymbol{\psi})]. \end{aligned} \tag{3.17}$$

The minimum average delay problem for the BE service in (3.11) can also be formulated as an MDP problem as follows.

$$\min_{\Omega_{BE}} \mathbb{E}^{\Omega_{BE}}[g(\boldsymbol{\psi}, \Omega_{DS}(\boldsymbol{\psi}), \boldsymbol{\psi}')], \tag{3.18}$$

where the per-stage delay cost is given as

$$g(\boldsymbol{\psi}, \Omega_{BE}(\boldsymbol{\psi}), \boldsymbol{\psi}') = \sum_n Q'_{Ln}. \tag{3.19}$$

Thus the optimal policy for the BE packets and the minimum average delay cost θ'^* can be derived by solving the following Bellman equation with relative value iteration.

$$\theta' + V(\boldsymbol{\psi}) = \min_{\Omega_{BE}} \sum_{\boldsymbol{\psi}'} Pr[\boldsymbol{\psi}'|\boldsymbol{\psi}, \Omega_{BE}(\boldsymbol{\psi})][g(\boldsymbol{\psi}, \Omega_{BE}(\boldsymbol{\psi}), \boldsymbol{\psi}') + V(\boldsymbol{\psi}')]. \tag{3.20}$$

For the system model defined, the total number of the system states of the DS and BE services are $(2(K_H + 1))^N$ and $(2(K_H + 1)(K_L + 1))^N$, respectively. If we use relative value iteration to solve the MDP problems in (3.15) and (3.20) and assume i and j iterations are needed to get the optimal policies, respectively, then the computational complexities for solutions to (3.15) and (3.20) are $O(iC_HN(2(K_H+1))^N)$ and $O(jC_LN(2(K_H+1)(k_L+1))^N)$, respectively [59], where C_H (C_L) is the average number of nonzero entries per row of (3.7) ((3.8)) with $4 < C_H < 6$ ($12 < C_L < 18$).

3.4 Online R-learning Algorithm

In section 3.3, the MDP formulation is based on the assumption that both traffic and channel characteristics are known. However, in practice, such information may not be always available. To address this issue, in this section, we remove this assumption and introduce a reinforcement learning (RL) algorithm, called R-learning, to solve the MDP problem only based on perfect observations of the channels' ON/OFF states.

Let c(x,a) be the immediate cost incurred by action a at state x. Since the traffic characteristics are unknown, we define the immediate cost $c(x,a) = \sum_n Q_{Hn}$ if $x \in \boldsymbol{\chi}$, and $c(x,a) = \sum_n Q_{Ln}$ if $x \in \boldsymbol{\psi}$, for the DS and BE packets, respectively. Instead of computing transition matrices and cost functions directly, R-learning is based on adaptive iterative learning of the action value function $R_t(x,a)$ and the average cost ρ. Here, $R_t(x,a)$ represents the average adjusted value of doing an action a at state x, and then following policy Ω in the future steps. It can be defined as [57]:

$$R^{\Omega}(x,a) = c(x,a) - \rho^{\Omega} + \sum_{x'} Pr[x'|x,a]V^{\Omega}(x'), \tag{3.21}$$

where $V^{\Omega}(x') = \min_{a\in A} R^{\Omega}(x')$, and ρ^{Ω} is the average cost under policy Ω.

The R-learning algorithm for solving (3.13) and (3.18) consists of the following steps:

1. At time $t=1$, initialize all the values $R_t(x,a)$ (e.g., 0) and average cost $\rho = 0$. Let x denote the current state.

2. Choose non-exploratory action a according to (3.24) with probability θ, whereas with probability $1-\theta$, choose exploratory action a uniformly from action space A.

3. Carry out action a. Let the next state be x'. Update R values using the following rule

$$R_{t+1}(x,a) = R_t(x,a)(1-\eta_r^t) + \eta_r^t[c(x,a) - \rho_t + \min_{a\in A} R_t(x',a)]. \tag{3.22}$$

If the non-exploratory action is selected, the average cost ρ is updated based on:

$$\rho_{t+1} = \rho_t(1-\eta_\rho) + \eta_\rho[c(x,a) + \min_{a\in A} R_t(x',a) - \min_{a\in A} R_t(x,a)]. \tag{3.23}$$

4. Set current state as x' and $t = t+1$, and go to step 2.

In (3.22) and (3.23), $0 \leq \eta_r^t \leq 1$ and $0 \leq \eta_\rho \leq 1$ are the learning rates of the action value $R(x,a)$ and the average cost ρ. They represents how quickly the errors in the estimated values are corrected.

We apply UE counter-based strategy as the exploration strategy. In this strategy, the non-exploratory action a is picked according to the following equation.

$$a = \arg\min R_t(x,a) + \frac{c}{N_t(x,a)}, \tag{3.24}$$

where c is a constant, and $N_t(x,a)$ represents the number of times that action a has been tried in state x till time t. With probability $1-\theta$, a random action is selected. The learning rate η_ρ^t for updating a particular $R(x,a)$ value is calculated as follows:

$$\eta_r^t(x,a) = \frac{\eta_0 k}{k + N_t(x,a)}, \tag{3.25}$$

where η_0 is the initial value of the η_r. According to (3.25), η_r is decayed based on the number of updates of a particular $R(x,a)$ value. Convergence is obtained after all the state-action pairs are visited infinitely often. After the convergence, the optimal decision at state x is $\arg\min_{a\in A} R(x,a)$.

3.5 A Myopic Method

Since the cardinality of the state space increases exponentially with the number of channels, directly solving MDP suffers from the well-known "curse of dimensionality". In addition, the computational complexity of R-learning is also very high with the increase of system states because of the convergence condition. From an implementation perspective, it is desirable to design a less-complex method with reasonable performance provisioning. In this section, a myopic method is proposed to solve the delay minimum problems defined in (3.15) and (3.20) with significantly reduced complexity.

Assume a DS packet J is sent to channel n by action a_n at time t and the current system state of channel n is $\chi_n(t) = (Q_{Hn}(t), S_n(t))$. We also assume that a decision and its action can be completed instantly. Then, after the arrival of the packet, the system state of channel n changes to $(Q_{Hn}(t)+1, S_n(t))$ immediately. We show that the sojourn time (i.e., the total time the packet spends in the system) for packet J follows phase type distribution. Define an absorbing Markov chain to represent the service process of packet J with $Q_{Hn} + 1$ transient states $\{1, 2, \ldots, Q_{Hn} + 1\}$ and one absorbing state $\{0\}$. Due to the FCFS policy, only the packets that are ahead of packet J get served first, and hence the transient state transition matrix becomes

$$\mathbf{T} = \begin{bmatrix} A_1 & & & \\ A_2 & A_1 & & \\ & \ddots & \ddots, & \\ & & A_2 & A_1 \end{bmatrix},$$

where $\mathbf{A_1} = \begin{bmatrix} \bar{q}_n P_{00} & \bar{q}_n P_{01} \\ P_{10} & P_{11} \end{bmatrix}$ and $\mathbf{A_2} = \begin{bmatrix} q_n P_{00} & q_n P_{01} \\ 0 & 0 \end{bmatrix}$. Note that T is a substochastic matrix. In addition, since the packets arriving later than packet J will not affect its sojourn time, the packet arrival is not considered in the transition process. Block A_1 denotes that the queue length remains the same, or there is no packet leaving. Block A_2 denotes the transition probability that the number of DS packets in the queue decreases by 1, which happens when one packet leaves the queue.

Let $\boldsymbol{t} = \boldsymbol{e}_1 - \boldsymbol{T}\boldsymbol{e}_1$, where $\boldsymbol{e}_1$ is a $2(Q_{Hn}+1)$-dimensional column vector of ones. Denote y as the initial vector, where the n-th element represents the probability that the system starts from a transient state n. Then, $\boldsymbol{y}$ is a row vector with a dimension of $2(Q_{Hn}+1)$. In this case, $\boldsymbol{y}=[0,0,\ldots,1,0]$ when the channel state is idle or $\boldsymbol{y}=[0,0,\ldots,0,1]$ when the channel state is busy. Thus, the probability that the sojourn time has a duration b_i can be calculated as

$$b_i = \boldsymbol{y}\boldsymbol{T}^{i-1}\boldsymbol{t},\ i \geq 1. \tag{3.26}$$

From (3.26), the mean sojourn time for packet J at channel n equals:

$$h(\boldsymbol{\chi}_n(t)) = \boldsymbol{y}(I-T)^{-2}\boldsymbol{t}, \tag{3.27}$$

where I is an identity matrix of size $2(Q_{Hn}+1)$.

The average sojourn time for each arriving BE packet can be derived in a similar way. Assume a BE packet is assigned to channel n after observing the system state $\boldsymbol{\psi}(t)$. Then, the service process of the BE packet can also be formulated to be a phase type distribution with transient states $\{1,2,\ldots,Q_{Ln}+1\}$ and an absorbing state $\{0\}$. The substochastic transition matrix becomes:

$$\mathbf{T'} = \begin{bmatrix} A'_1 & & & \\ A'_2 & A'_1 & & \\ & \ddots & \ddots, & \\ & & A'_2 & A'_1 \end{bmatrix},$$

where A'_0, A'_1 and A'_2 are defined in section 3.2 with $a_H(n) = p_{cH}(n),\, b_L(n) = q_L(n),\, b_H(n) = q_H(n), a_L(n) = 0$. Therefore, the mean sojourn time for the arriving packet is:

$$h(\boldsymbol{\psi}, A'(t)) = \boldsymbol{y}'(I-T')^{(-2)}\boldsymbol{t}', \tag{3.28}$$

where $\boldsymbol{t}' = \boldsymbol{e}_2 - \boldsymbol{T}'\boldsymbol{e}_2$, $\boldsymbol{e}_2$ is a $2(Q_{Ln}+1)(K_H+1)$-dimensional column vector of ones, and $\boldsymbol{y}'$ is a $2(Q_{Ln}+1)(K_H+1)$-dimensional row vector with element 1 at the start state and 0 anywhere else.

Note that (3.14) and (3.19) define the per-stage delay costs based on the assumption that the channel selection probabilities are known. In addition, the mean sojourn times in (3.27) and (3.28) are online estimations for the delay of each arriving packet based on both the current system state and the possible future queueing dynamics.

We can now define the proposed myopic policy, which aims at minimizing the immediate cost instead of considering the impact of the current action on the future. Specifically, at each time slot, we first consider channel assignment for the arriving DS packet. We calculate the mean sojourn time of each channel based on (3.27) and select the channel that has the minimum value. We update the channel selection probability vector $P = \{p_1, \ldots, p_N\}$ according to $p_n = \frac{Num(n)}{Sum}$, where $Num(n)$ is the number of slots that channel n is selected for the DS packets, and Sum is the total number of time slots so far. After that, we compute the mean sojourn time of each channel for the BE packets using (3.28) and select the channel with the minimum mean sojourn time for the newly arrived BE packet. The myopic algorithm for the channel selection is summarized in Algorithm 1.

The computation complexities of the myopic algorithm are $O((2(Q_{Hn} + 1))^3)$ and $O((2(Q_{Ln} + 1)(K_H + 1))^3)$ at each time slot for the DS and BE packets, respectively. The calculation mainly considers the computation complexity of (3.27) and (3.28). Obviously, these complexities are much smaller than MDP solutions. In addition, the myopic algorithm needs little storage space while a large storage space with size $(2(K_H + 1)(k_L + 1))^N + (2(K_H + 1))^N$ is needed for solving the MDP problem by the relative value iteration.

3.6 Simulation Results

In this section, we compare the delay performance of the decision policies obtained from the proposed schemes for both DS and BE packets via Matlab based simulation results. For the case with full knowledge about the environment, we obtain the optimal policy by solving the MDP problems using the relative value iteration (RVI).

Algorithm 1 Myopic Channel Selection algorithm

```
procedure ONLINE CHANNEL SELECTION
Initialization: $N_m := 0, Sum := 0, P := 0, p_{cH}(1) := 1, p_{cH}(n) := 0, n = 1, 2, 3 \ldots, N$.
    for each time slot $t$ do
        for the arriving DS packets do
        Obtain the system state information $\boldsymbol{\chi}(t) = (\boldsymbol{Q}_H(t), \boldsymbol{S}(t))$.
            for each channel $n \in N$ do
            Update $y, T, t$ base on $Q_{Hn}(t), S_n(t)$.
            Compute the mean sojourn time for channel $n$ using (3.27).
            end for
        Select the channel with the minimum mean sojourn time.
        end for
        for $n = 1 : N$ do
            if the $n$-th channel is chosen then
            $Num(n) = Num(n) + 1$;
            end if
        $Sum = Sum + 1$;
        $p_n = \frac{Num(n)}{Sum}$;
        $p_{cH}(n) = p \times p_n$
        end for
        for the arriving BE packets do
        Obtain the system state information $\boldsymbol{\psi}(t) = (\boldsymbol{Q}_H(t), \boldsymbol{Q}_L(t), \boldsymbol{S}(t))$.
            for each channel $n \in N$ do
            Update $y', T', t'$ base on $Q_{Ln}(t), Q_{Hn}(t), S_n(t)$.
            Compute the mean sojourn time for channel $n$ using (3.28).
            end for
        Select the channel with the minimum mean sojourn time.
        end for
    end for
end procedure
```

In the simulations, a CRN with one SU and N PUs is configured, where N is set to 3. The service rates for the SU's packets and the PU's packet at channel $1, 2$ and 3 are $b_{H1} = b_{L1} = 0.8$,$b_{H2} = b_{L2} = 0.7, b_{H3} = b_{L3} = 0.4$, and $\beta_1 = 0.85, \beta_2 = 0.75, \beta_3 = 0.55$, respectively. The default arrival rates of the packets of PU1, PU2, PU3 and SU's DS packets are $\alpha_1 = 0.15, \alpha_2 = 0.1, \alpha_3 = 0.1$ and $p_H = 0.2$, respectively. For each simulation result, one or two of these parameters vary at a time while others are kept same as default values. The buffer lengths are set to $K_p = 10, K_H = 4$, and $K_L = 5$. Note that due to the high computational complexity of the optimal MDP schemes, some simulation parameters are set at small values to allow the simulation feasible. However, the observations are general to other settings of simulation parameters. The parameters for the R-learning algorithm are set as: $c = 100, \eta_0 = 0.5, k = 500$ and $\eta_\rho = 0.05$. The policy for the BE service is carried out based on the cumulative reward obtained over 2,000 runs of 500,000 steps in the R-learning algorithm. For comparison purpose, another spectrum decision method called the shortest queue channel selection scheme is also simulated, where both the DS and BE packets select the channel with the shortest queue length for transmission.

In Fig. 3.2, the delay performance of the proposed schemes is evaluated for the DS packets. It can be seen that as the DS packets' arrival rate increases, the average delay increases. The simulation results of the optimal policy obtained by RVI can well match the analytical results, which justifies the accuracy of our proposed analytical model. The R-learning algorithm converges to the same optimal solution as RVI, while the performance of the myopic algorithm is slightly worse than them. However, as the DS packets' arrival rate increases, the outperformance of the proposed schemes become more obvious compared to the shortest queue scheme. This is because the shortest queue scheme only considers the current queue length, which can not accurately represent the delay that the arriving packet is actually going to experience, while the myopic scheme estimates the packet delay which is more accurate for making decision. The MDP schemes perform best because it considers both the current system state and the influence of the current channel selection on the future packets.

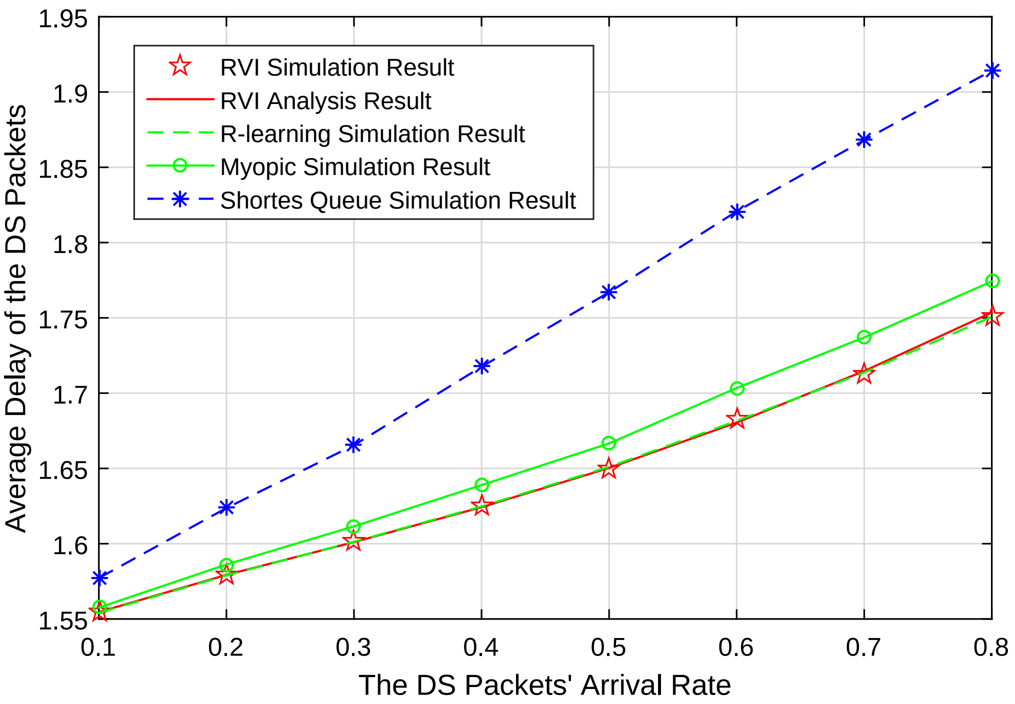

Figure 3.2: Average Delay for the DS packets with increasing arrival rate

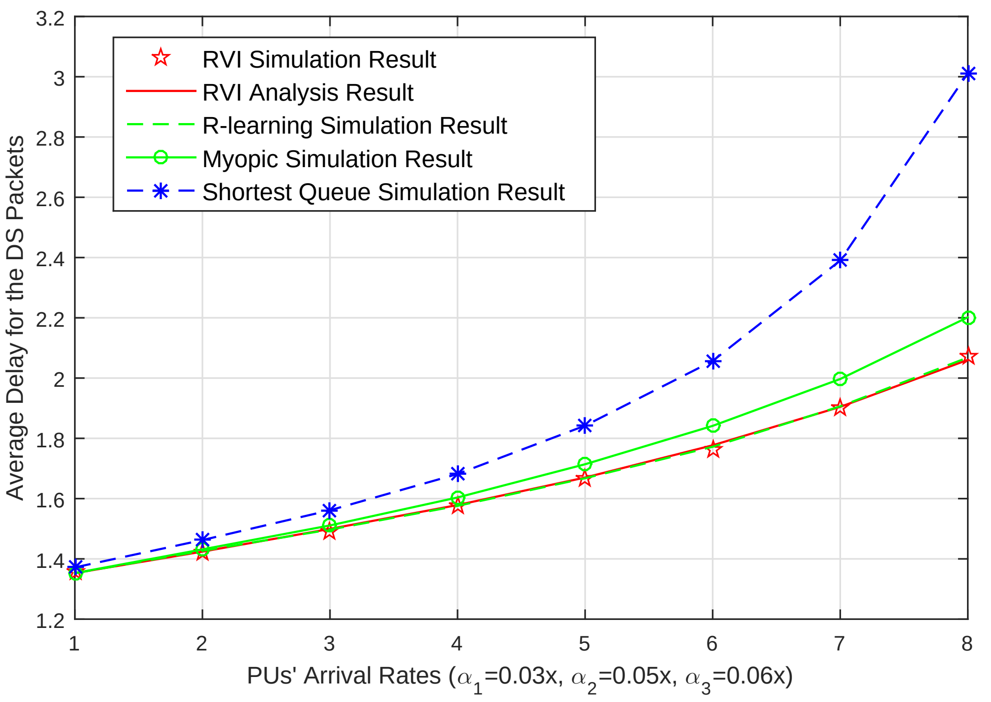

Figure 3.3: Average Delay for the DS packets with increasing PUs arrival rate

In Fig. 3.3, the delay performance of the DS packets is evaluated when the PU's arrival rate changes. In our simulation, the arrival rates of PU1, PU2 and PU3 are set to $0.03x, 0.04x, 0.06x\ (x = 1, \ldots, 8)$, respectively. From the figure, it can be seen that the average delay of the DS packets increases as PUs' arrival rate increases. This is because as PUs' arrival rates increase, the channels' idle probabilities decrease, which leads to longer delay for the SU's packets.

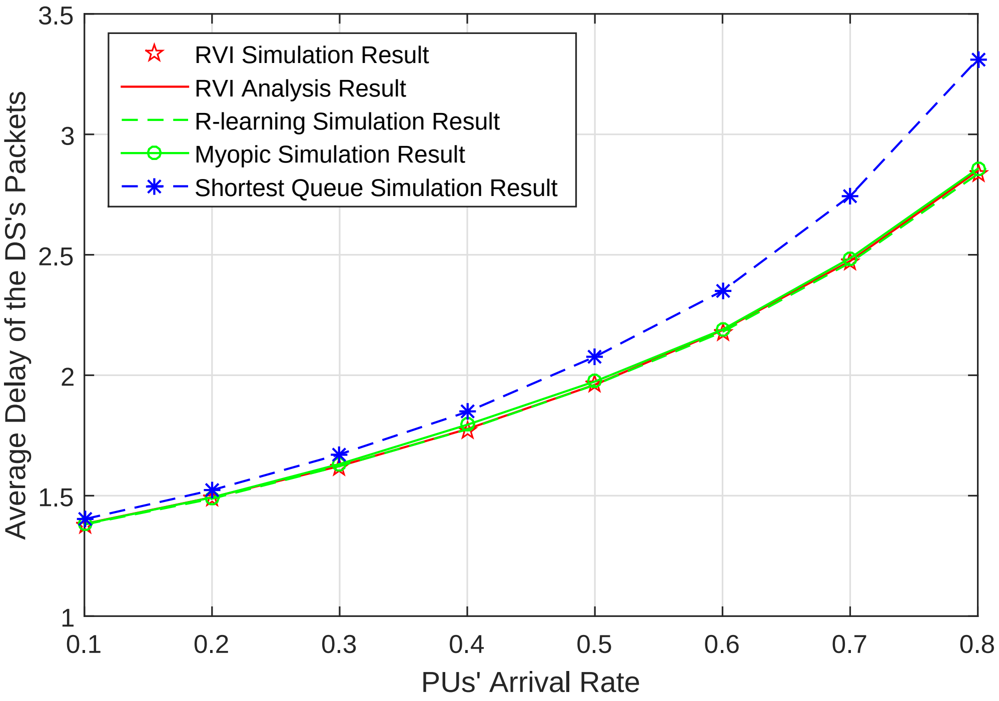

Figure 3.4: Average Delay for the DS packets with increasing PUs arrival rate (Three channels)

We repeat a similar simulation as shown in Fig. 3.4, where all the PUs' arrival rates are set to be the same. Comparing Figs. 3.3 and 3.4, we can observe that the performance gap between the shortest queue scheme and the proposed schemes under the heterogenous channel conditions is larger than the homogeneous case. It is because when the channel states become more heterogeneous, the influence of channel selection decision on the future becomes stronger so that the schemes with the consideration of future work better.

In Fig. 3.5, the simulation is carried out under a two-channel scenario and homogeneous PUs' arrival rates are considered. Compared to Fig. 3.4, it can be seen that the performance

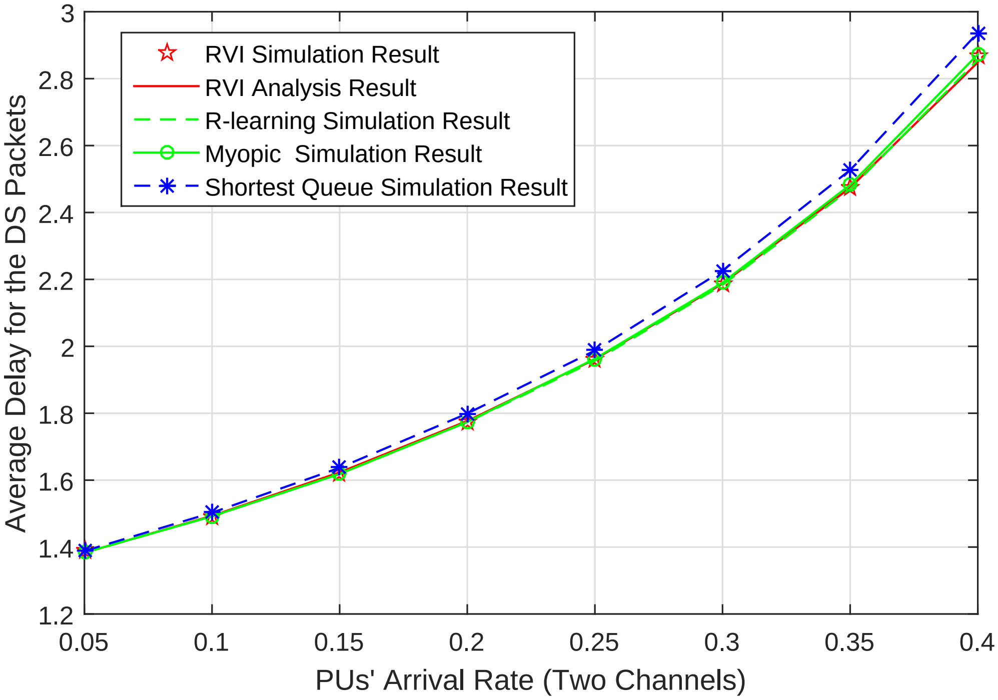

Figure 3.5: Average Delay for the DS packets with increasing PUs arrival rate (Two channels)

of the three methods becomes quite similar in this simulation. The reason is that the more channels exist, the more heterogeneous the system may become and the more advantage the MDP schemes may have.

Most importantly, from all the above figures, we can see that the performance of the myopic algorithm is slightly worse compared to the optimum based on MDP, but is much better than the shortest queue scheme. It is because the myopic method considers the current decision on the future network evolution to some extend. By considering its low computational complexity, the myopic scheme is more suitable for practical applications.

The delay performance of the proposed schemes for the BE packets are shown in Fig. 3.6. Similarly, the average delay of the BE packets increases as their arrival rate increases and the MDP schemes perform the best among the three methods. It further justifies the fact that the MDP schemes allocate the channel resources in the most efficient way among all the schemes, so that both DS and BE services achieve the minimum delay. The myopic scheme performs better than the shortest queue scheme when the BE packets' arrival rate

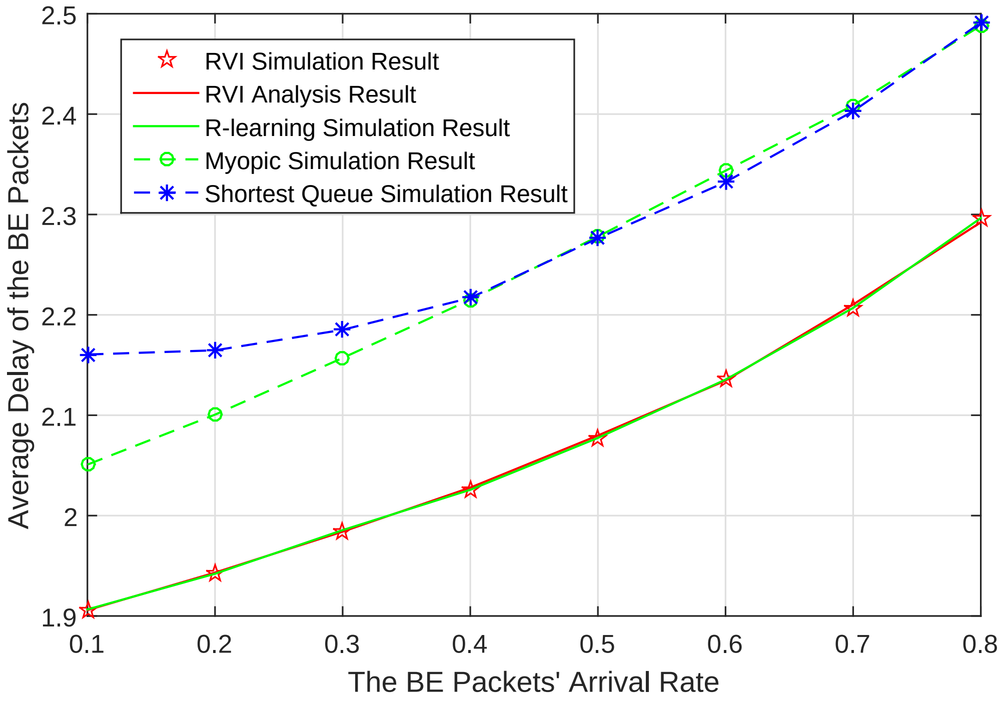

Figure 3.6: Average Delay for the BE packets with increasing arrival rate

is small and two schemes achieve almost equal performance when the BE packets' arrival rate becomes large. In addition, the performance of the myopic scheme is much worse than the optimum. The reason is that more resources are allocated to the DS services under the myopic scheme, and hence as the arrival rate increases, the channel resources may become scarce so as to cause larger delay to BE services.

Chapter 4

Conclusions and Future Works

4.1 Conclusions

In this thesis, we firstly reviewed the characteristics and functions of CRNs. More specifically, we introduced the general spectrum decision framework for CRNs and provided a comprehensive literature review on channel selection. In chapter 2, we explained some fundamental knowledge of MDP and RL which was used in chapter 3.

In chapter 3, channel selection policies were derived for the SU with multiple-class services, based on which an optimal channel was selected upon each packet's arrival. A priority queue model with an ON/OFF server was proposed to analyze the evolvement of queue lengths at each channel, with heterogenous PUs' activities, channel conditions and SU's packets arrival rates considered. Based on the queueing analysis, an MDP method was developed to obtain the minimum delay policies, according to which the SU's DS and BE services selected the best transmission channel for each packet. The problem in the case with no prior knowledge about the environment was also solved using R-learning algorithm. Simulation results showed that based on partial information the R-learning algorithm could converge to the optimal result. To reduce complexity of solving the MDP problems directly, we proposed a myopic algorithm where a decision was made based on the estimated delay of the current packet. Simulation results also showed that the MDP schemes greatly improved the delay performance for both DS and BE services compared to the shortest queue scheme, and the myopic scheme was able to achieve a better performance for the DS services with a cost of longer delay for the BE services.

4.2 Future Works

In this section, some practical extensions, challenges that may occur for the extensions and possible solutions are discussed for future research.

In our proposed channel selection scheme, the DS service is granted a higher priority over the BE services on all available channels. However, in some applications, the BE service may also have delay requirements even though not as strict as DS service. To ensure delay requirements for the BE service, certain portion of available channels can be reserved so that these channels will allow the BE service to have higher access priority than the DS service. In this scheme, the decision policy for the BE packets will affect the decision making for DS packets and thus spectrum decisions for both services will be interacted which makes the channel selection problems difficult. Furthermore, the system model can be extended to a CRN in which multiple SUs have heterogeneous packet arrival distributions. Packets from different SUs with a same priority will be sent to the same queue on each channel. Each SU performs as an independent decision maker to determine channel selections. Since each SU's spectrum decision depends on the other SUs' decisions, the problem becomes even more complex.

For both of the above extensions, reinforcement learning method can be applied to find channel selection policies. In the first extension, the DS and BE services can learn each other's channel selection policies by interacting with the environment and adjust their decisions. In the second extension, through reinforcement learning each SU learns the other SUs' channel selection strategies and arrival rates on each channel so that it can estimate the average delay and make spectrum decisions.

In our current work, we only consider that the channel coefficients are i.i.d. among time slots. To be more practical, channel coefficients can be considered to be correlated among time slots which will leads to correlated service rates. Therefore the number of channel states will increase and the size of transition matrices will be greatly increased. As a result, the computational complexity will be hugely increased as well. To address this issue MDP approximation algorithms with state space reduction may be employed.

Full knowledge of all channel states is assumed in our work. However, acquiring such knowledge is energy-hungry and hard-ware demanding, and thus low-cost and battery-powered wireless nodes may have to employ partial spectrum monitoring (i.e., only sensing part of channels). On the other hand, it is usually difficult for a real sensing equipment to implement perfect sensing, which inevitably leads to some sensing errors. Under partial spectrum monitoring and imperfect sensing, system states cannot be fully perfectly observed. Our problem then becomes one of the POMDPs with uncertain channel states, which is very challenging since POMDPs are often computationally intractable to solve optimally. Again, some approximation methods suitable for this specific problem have to be developed to reduce the computational complexity.

Reference

[1] P. Kolodzy and I. Avoidance, “Spectrum policy task force,” *Federal Commun. Comm., Washington, DC, Rep. ET Docket*, no. 02-135, 2002.

[2] Q. Zhao and B. Sadler, “A survey of dynamic spectrum access,” *Signal Processing Magazine, IEEE*, vol. 24, no. 3, pp. 79–89, May 2007.

[3] J. Peha, “Sharing spectrum through spectrum policy reform and cognitive radio,” *Proceedings of the IEEE*, vol. 97, no. 4, pp. 708–719, April 2009.

[4] I. Akyildiz, W.-Y. Lee, M. C. Vuran, and S. Mohanty, “A survey on spectrum management in cognitive radio networks,” *Communications Magazine, IEEE*, vol. 46, no. 4, pp. 40–48, 2008.

[5] F. C. Commission, “Fcc adopts rules for unlicensed use of television white spaces,” *Media Release, USA, Accessed: 18/04/2011. [Online]. Available: http://transition.fcc.gov/sptf/headlines2008.html*, Nov. 4 2008.

[6] O. of Communications, “Statement on cognitive access to interleaved spectrum,” *United Kingdom, Accessed: 18/04/2011. [Online]. Available: http://stakeholders.ofcom.org.uk/binaries/consultations/ cognitive/ statement/statement.pdf*, Jul. 2009.

[7] I. F. Akyildiz, W.-Y. Lee, M. C. Vuran, and S. Mohanty, “Next generation/dynamic spectrum access/cognitive radio wireless networks: a survey,” *Computer networks*, vol. 50, no. 13, pp. 2127–2159, 2006.

[8] I. Balapuwaduge, L. Jiao, V. Pla, and F. Li, “Channel assembling with priority-based queues in cognitive radio networks: Strategies and performance evaluation,” *Wireless Communications, IEEE Transactions on*, vol. 13, no. 2, pp. 630–645, February 2014.

[9] J. Wang, H. Zhai, and Y. Fang, “Opportunistic packet scheduling and media access control for wireless lans and multi-hop ad hoc networks,” in *Wireless Communications and Networking Conference, 2004. WCNC. 2004 IEEE*, vol. 2, March 2004, pp. 1234–1239 Vol.2.

[10] Y. Wu, F. Hu, S. Kumar, Y. Zhu, A. Talari, N. Rahnavard, and J. Matyjas, “A learning-based qoe-driven spectrum handoff scheme for multimedia transmissions over cognitive radio networks,” *Selected Areas in Communications, IEEE Journal on*, vol. 32, no. 11, pp. 2134–2148, November 2014.

[11] H.-P. Shiang and M. van der Schaar, “Queuing-based dynamic channel selection for heterogeneous multimedia applications over cognitive radio networks,” *Multimedia, IEEE Transactions on*, vol. 10, no. 5, pp. 896–909, Aug 2008.

[12] F. C. Commission, “Notice of proposal rulemaking and order,” *ET Docket, USA*, pp. 03–322, Nov. 2003.

[13] I. Akyildiz, W.-Y. Lee, M. C. Vuran, and S. Mohanty, “A survey on spectrum management in cognitive radio networks,” *Communications Magazine, IEEE*, vol. 46, no. 4, pp. 40–48, 2008.

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

[15] C. Santivanez, R. Ramanathan, C. Partridge, R. Krishnan, M. Condell, and S. Polit, “Opportunistic spectrum access: Challenges, architecture, protocols,” in *Proceedings of the 2Nd Annual International Workshop on Wireless Internet*, ser. WICON ’06. New York, NY, USA: ACM, 2006. [Online]. Available: http://doi.acm.org/10.1145/1234161.1234174

[16] M. Nekovee, “A survey of cognitive radio access to tv white spaces,” in *Ultra Modern Telecommunications Workshops, 2009. ICUMT ’09. International Conference on*, Oct 2009, pp. 1–8.

[17] M. Masonta, M. Mzyece, and N. Ntlatlapa, “Spectrum decision in cognitive radio networks: A survey,” *Communications Surveys Tutorials, IEEE*, vol. 15, no. 3, pp. 1088–1107, Third 2013.

[18] J. Wang, M. Ghosh, and K. Challapali, “Emerging cognitive radio applications: A survey,” *Communications Magazine, IEEE*, vol. 49, no. 3, pp. 74–81, March 2011.

[19] M. Hoyhtya, S. Pollin, and A. Mammela, “Improving the performance of cognitive radios through classification, learning, and predictive channel selection,” *Advances in Electronics and Telecommunications*, vol. 2, no. 4, pp. 28–38, 2011.

[20] B. Canberk, I. Akyildiz, and S. Oktug, “Primary user activity modeling using first-difference filter clustering and correlation in cognitive radio networks,” *Networking, IEEE/ACM Transactions on*, vol. 19, no. 1, pp. 170–183, Feb 2011.

[21] Q. Zhao, L. Tong, A. Swami, and Y. Chen, “Decentralized cognitive mac for opportunistic spectrum access in ad hoc networks: A pomdp framework,” *Selected Areas in Communications, IEEE Journal on*, vol. 25, no. 3, pp. 589–600, April 2007.

[22] Q. Zhao, B. Krishnamachari, and K. Liu, “On myopic sensing for multi-channel opportunistic access: structure, optimality, and performance,” *Wireless Communications, IEEE Transactions on*, vol. 7, no. 12, pp. 5431–5440, December 2008.

[23] S. Ahmad, M. Liu, T. Javidi, Q. Zhao, and B. Krishnamachari, “Optimality of myopic sensing in multichannel opportunistic access,” *Information Theory, IEEE Transactions on*, vol. 55, no. 9, pp. 4040–4050, Sept 2009.

[24] A. Chronopoulos, M. Musku, S. Penmatsa, and D. Popescu, “Spectrum load balancing for medium access in cognitive radio systems,” *Communications Letters, IEEE*, vol. 12, no. 5, pp. 353–355, May 2008.

[25] I. Malanchini, M. Cesana, and N. Gatti, “On spectrum selection games in cognitive radio networks,” in *Global Telecommunications Conference, 2009. GLOBECOM 2009. IEEE*, Nov 2009, pp. 1–7.

[26] R. Xie, F. Yu, and H. Ji, "Dynamic resource allocation for heterogeneous services in cognitive radio networks with imperfect channel sensing," *Vehicular Technology, IEEE Transactions on*, vol. 61, no. 2, pp. 770–780, Feb 2012.

[27] B. Awoyemi, B. Maharaj, and A. Alfa, "Resource allocation for heterogeneous cognitive radio networks," in *Wireless Communications and Networking Conference (WCNC), 2015 IEEE*. IEEE, 2015, pp. 1759–1763.

[28] D.-H. Na, H. Nan, and S.-J. Yoo, "Policy-based dynamic channel selection architecture for cognitive radio networks," in *Communications and Networking in China, 2007. CHINACOM '07. Second International Conference on*, Aug 2007, pp. 1190–1194.

[29] T. Jiang, H. Wang, and A. Vasilakos, "Qoe-driven channel allocation schemes for multimedia transmission of priority-based secondary users over cognitive radio networks," *Selected Areas in Communications, IEEE Journal on*, vol. 30, no. 7, pp. 1215–1224, August 2012.

[30] N. Motamedi, S. Kumar, F. Hu, and N. Rowe, "A priority-aware channel selection scheme for real-time data transmission in cognitive radio networks," in *Computing, Networking and Communications (ICNC), 2013 International Conference on*, Jan 2013, pp. 734–739.

[31] M. Hong, J. Kim, H. Kim, and Y. Shin, "An adaptive transmission scheme for cognitive radio systems based on interference temperature model," in *Consumer Communications and Networking Conference, 2008. CCNC 2008. 5th IEEE*, Jan 2008, pp. 69–73.

[32] Y. Che, J. Wang, J. Chen, W. Tang, and S. Li, "Hybrid power control scheme in hierarchical spectrum sharing network for cognitive radio," *Physical Communication*, vol. 2, no. 1, pp. 73–86, 2009.

[33] W. Ren, Q. Zhao, and A. Swami, "Power control in spectrum overlay networks: how to cross a multi-lane highway," in *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*. IEEE, 2008, pp. 2773–2776.

[34] T. Weingart, D. C. Sicker, and D. Grunwald, "A statistical method for reconfiguration of cognitive radios," *Wireless Communications, IEEE*, vol. 14, no. 4, pp. 34–40, 2007.

[35] B. Kasiri, J. Cai, A. Alfa, and W. Wang, "A distributed cooperative attack on the multi-channel spectrum sensing: A coalitional game study," in *Global Telecommunications Conference (GLOBECOM 2011), 2011 IEEE*, Dec 2011, pp. 1–5.

[36] S. Ahmad and M. Liu, "Multi-channel opportunistic access: A case of restless bandits with multiple plays," in *Communication, Control, and Computing, 2009. Allerton 2009. 47th Annual Allerton Conference on*, Sept 2009, pp. 1361–1368.

[37] F. F. Digham, "Joint power and channel allocation for cognitive radios," in *Wireless Communications and Networking Conference, 2008. WCNC 2008. IEEE*. IEEE, 2008, pp. 882–887.

[38] T. Shu and M. Krunz, "Exploiting microscopic spectrum opportunities in cognitive radio networks via coordinated channel access," *Mobile Computing, IEEE Transactions on*, vol. 9, no. 11, pp. 1522–1534, 2010.

[39] Y. Yao and Z. Feng, "Centralized channel and power allocation for cognitive radio networks: a q-learning solution," in *Future Network and Mobile Summit, 2010*. IEEE, 2010, pp. 1–8.

[40] W. Wang, K. G. Shin, and W. Wang, "Joint spectrum allocation and power control for multihop cognitive radio networks," *Mobile Computing, IEEE Transactions on*, vol. 10, no. 7, pp. 1042–1055, 2011.

[41] L.-C. Wang, C.-W. Wang, and F. Adachi, "Load-balancing spectrum decision for cognitive radio networks," *Selected Areas in Communications, IEEE Journal on*, vol. 29, no. 4, pp. 757–769, 2011.

[42] C. T. Do, N. H. Tran, C. S. Hong, S. Lee, J.-J. Lee, and W. Lee, "A lightweight algorithm for probability-based spectrum decision scheme in multiple channels cognitive radio networks," *Communications Letters, IEEE*, vol. 17, no. 3, pp. 509–512, 2013.

[43] F. Sheikholeslami, M. Nasiri-Kenari, and F. Ashtiani, "Optimal probabilistic initial and target channel selection for spectrum handoff in cognitive radio networks," *Wireless Communications, IEEE Transactions on*, vol. 14, no. 1, pp. 570–584, 2015.

[44] A. El Shafie and T. Khattab, "On orthogonal band allocation for multiuser multiband cognitive radio networks: Stability analysis," *Communications, IEEE Transactions on*, vol. 63, no. 1, pp. 37–50, 2015.

[45] H.-P. Shiang and M. Van der Schaar, "Queuing-based dynamic channel selection for heterogeneous multimedia applications over cognitive radio networks," *Multimedia, IEEE Transactions on*, vol. 10, no. 5, pp. 896–909, 2008.

[46] Y. Wu, F. Hu, S. Kumar, Y. Zhu, A. Talari, N. Rahnavard, and J. D. Matyjas, "A learning-based qoe-driven spectrum handoff scheme for multimedia transmissions over cognitive radio networks," *Selected Areas in Communications, IEEE Journal on*, vol. 32, no. 11, pp. 2134–2148, 2014.

[47] Y. Zhao, M. Song, and C. Xin, "Delay analysis for cognitive radio networks supporting heterogeneous traffic," in *Sensor, Mesh and Ad Hoc Communications and Networks (SECON), 2011 8th Annual IEEE Communications Society Conference on*. IEEE, 2011, pp. 215–223.

[48] V. K. Tumuluru, P. Wang, D. Niyato, and W. Song, "Performance analysis of cognitive radio spectrum access with prioritized traffic," *Vehicular Technology, IEEE Transactions on*, vol. 61, no. 4, pp. 1895–1906, 2012.

[49] Y. Cui, V. K. Lau, R. Wang, H. Huang, and S. Zhang, "A survey on delay-aware resource control for wireless systemslarge deviation theory, stochastic lyapunov drift, and distributed stochastic learning," *Information Theory, IEEE Transactions on*, vol. 58, no. 3, pp. 1677–1701, 2012.

[50] Z. Hou, J. A. Filar, and A. Chen, *Markov processes and controlled Markov chains*. Springer Science & Business Media, 2013.

[51] E. R. Zieyel, "Operations research: Applications and algorithms," *Technometrics*, vol. 30, no. 3, pp. 361–362, 1988.

[52] D. P. Bertsekas, *Dynamic programming and optimal control*. Athena Scientific Belmont, MA, 1995, vol. 1, no. 2.

[53] R. Howard, "Dynamic programming and markov decision processes," *Cambridge, MA*, 1960.

[54] A. Gosavi, “A variant of the relative value iteration algorithm for solving markov decision problems,” in *IIE Annual Conference. Proceedings*. Institute of Industrial Engineers-Publisher, 2002, p. 1.

[55] C. J. C. H. Watkins, “Learning from delayed rewards,” Ph.D. dissertation, University of Cambridge England, 1989.

[56] A. Schwartz, “A reinforcement learning method for maximizing undiscounted rewards,” in *Proceedings of the tenth international conference on machine learning*, vol. 298, 1993, pp. 298–305.

[57] S. Mahadevan, “Average reward reinforcement learning: Foundations, algorithms, and empirical results,” *Machine learning*, vol. 22, no. 1-3, pp. 159–195, 1996.

[58] A. S. Alfa, *Queueing theory for telecommunications: discrete time modelling of a single node system*. Springer Science & Business Media, 2010.

[59] O. Shlakhter, C.-G. Lee, D. Khmelev, and N. Jaber, “Acceleration operators in the value iteration algorithms for markov decision processes,” *Operations research*, vol. 58, no. 1, pp. 193–202, 2010.

Publication List

[1] **H. Tian**, J. Cai, A. S. Alfa, S. Huang, and H. Cao, ”Dynmaic load-balancing spectrum decision for cognitive radio networks with multi-class services,” *Wireless Communications & Signal Processing (WCSP), International Conference on. IEEE, 2015.* Nanjing, China, October 15-17, 2015.