2011

Spectrum Allocation Algorithms for Cognitive Radio Mesh Networks

Hisham Mohammad Almasaeid
Iowa State University

Spectrum allocation algorithms for cognitive radio mesh networks

by

Hisham Mohammad Almasaeid

A dissertation submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Major: Computer Engineering

Program of Study Committee:
Ahmed E. Kamal, Major Professor
Sarah Ryan
Sang Kim
Morris Chang
Daji Qiao

Iowa State University

Ames, Iowa

2011

DEDICATION

To my beloved parents...

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGMENTS

All praise is due to ALLAH, the most merciful and the most compassionate, for bestowing this great blessing on me. I thank the Almighty ALLAH for granting me the opportunity to pursue this Ph.D. degree, and to meet and connect with great people at both the personal and professional level.

I would like to express my sincere thanks and gratitude to those who helped me throughout this research. First and foremost, Dr. Ahmed E. Kamal, my supervisor, for his support, guidance, and inspiration throughout the period of my study. I would also like to thank my committee members, Professors Sarah Ryan, Sang Kim, Morris Chang, and Daji Qiao for their valuable time and advice. I also owe a great deal of thanks to my colleagues in the *Laboratory for Advanced Networks*, Ashraf Hammad, Osameh Al-Kofahi, Mohammad Saleh, Sharhabeel Alnabelsi, Ramzi Saifan, Long Long, Mirzad Mohandespour, and Tasneem Jawadwala.

I should also acknowledge and thank my great friends in the Ames community who made my life full of happiness and great memories. In particular, the Dar Al-Arqum Mosque family, Mohammad Al-Saqer, Nimer Mehyar, Mohammad Fraiwan, Bashar Gharaibeh, Naeem Al-Oudat, Haithem Al-Mofleh, Hajem Bataineh, Taha Abdelshafy, and Abdel Hamid Azzaz.

I am indebted to my dear cousins, Helal K. Alqunais, Salameh K. Alqunais, and Hassan H. Alqunais for their support. It would not have been possible for me to pursue this degree without their help. I would further like to thank Yarmouk University for sponsoring me throughout the course of my study, and the National Science Foundation (NSF) for financially supporting my research under grants CNS-0626822, ECS-060157, and CNS-072145. Finally, I would like to thank all my family members and friends for their love and support.

ABSTRACT

Empowered by the cognitive radio technology, and motivated by the sporadic channel utilization, both spatially and temporally, dynamic spectrum access networks (also referred to as cognitive radio networks and next generation wireless networks) have emerged as a solution to improve spectrum utilization and provide more flexibility to wireless communication. A cognitive radio network is composed of wireless users, referred to as secondary users, which are allowed to use a given licensed spectrum band as long as there are no primary, licensed, users occupying this band in their vicinity. This restricted spectrum access strategy leads to heterogeneity in channel availability among secondary users. Heterogeneity in channel availability forms a significant source of performance degradation for cognitive radio networks, and poses a great challenge on protocol design. In this dissertation, we propose spectrum allocation algorithms that take this heterogeneity property and its effect on the network performance into consideration.

The spectrum allocation solutions proposed in this dissertation address three major objectives in cognitive radio mesh networks. The first objective is maximizing the network coverage, in terms of the total number of served clients, and at the same time simplifying the communication coordination function. To address this objective, we proposed a received based channel allocation strategy that alleviates the need for a common control channel, thus simplifying the coordination function, and at the same time maximizes the number of clients served with link reliability guarantees. We show the superiority of the proposed allocation strategy over other existing strategies.

The second objective is improving the multicast throughput to compensate for the performance degradation caused by the heterogeneity in channel availability. We proposed a

scheduling algorithm that schedules multicast transmissions over both time and frequency and integrates that with the use of network coding. This algorithm achieves a significant gain, measured as the reduction in the total multicast time, as the simulation results prove. We also proposed a failure recovery algorithm that can adaptively adjust the schedule in response to temporary changes in channel availability.

The last objective is minimizing the effect of channel switching on the end-to-end delay and network throughput. Channel switching can be a significant source of delay and bandwidth wastage, especially if the secondary users are utilizing a wide spectrum band. To address this issue, we proposed an on-demand multicast routing algorithm for cognitive radio mesh networks based on dynamic programming. The algorithm finds the best available route in terms of end-to-end delay, taking into consideration the switching latency at individual nodes and the transmission time on different channels. We also presented the extensibility of the proposed algorithm to different routing metrics. Furthermore, a route recovery algorithm that takes into consideration the overhead of rerouting and the route cost was also proposed. The gain of these algorithms was verified by simulation.

CHAPTER 1. INTRODUCTION AND RESEARCH GOALS

Empowered by the cognitive radio technology and motivated by the sporadic channel utilization, both spatially and temporally, dynamic spectrum access networks (also referred to as cognitive radio networks and next generation wireless networks) have emerged as a solution to improve spectrum utilization and provide more flexibility to wireless communication. In this chapter, we first give a brief background about cognitive radio networks, including motivation and network architecture. Then, we layout the research issues related to cognitive radio networks that we consider in this dissertation. We then review the literature for existing work that addresses these issues.

1.1 Cognitive Radio Networks: background and motivation

The need for dynamic spectrum access was motivated by two main observations; spectrum crowdedness and spectrum underutilization. These two problems resulted from the current fixed channel assignment policy, like the one adopted by the Federal Communications Commission (FCC) in the United States. According to this policy, spectrum bands are assigned to some license holders or services on a long term basis. In most cases, the assigned spectrum bands are also reserved over vast geographical regions [1]. Therefore, unlicensed spectrum is becoming more scarce and insufficient to accommodate the increasing spectrum demand. Another disadvantage of the fixed channel assignment policy comes from the current spectrum usage pattern. FCC reports indicate that variations in the utilization of assigned spectrum bands range from 15% to 85% [1].

The enabling technology of dynamic spectrum access networks is the technology of cognitive radio [2, 3]. A cognitive radio is a software defined radio that is able to identify idle frequency channels, or so called spectrum holes, through *spectrum sensing*. Furthermore, a cognitive radio is able to identify the best available spectrum to meet certain quality-of-service requirements through *spectrum analysis*. *Spectrum mobility* is another functionality of a cognitive radio which enables it to change its frequency of operation, and therefore enables dynamic spectrum access. In general terms, a *"cognitive radio" is defined as a radio that can change its transceiver parameters based on interaction with the surrounding environment.*

These aforementioned functionalities of a cognitive radio form the basic characteristics of the new communication paradigm for dynamic spectrum access networks (cognitive radio networks). In this paradigm, wireless users are classified into two categories based on whether they are licensed to use a particular spectrum band or not, and those are primary, i.e, licensed, users (PUs) and secondary, i.e, unlicensed, users (SUs). SUs are allowed to opportunistically use the spectrum as long as they do not cause harmful interference to active PUs. Therefore, for an SU, the term *"available (or idle) channel"* is used to refer to a frequency channel that can be used by an SU such that no harmful interference will affect any nearby PU receiver. This is achievable if PU receivers are far enough from the SU transmitter, i.e., *spatial channel availability*, or no PU receivers are receiving while the SU transmitter is transmitting, i.e., *temporal channel availability*. This opportunistic and dynamic communication paradigm leads to higher spectrum utilization, and provides SUs with good service availability and reliability because they can access any part of the spectrum as long as they do not interrupt an ongoing PU transmission, and can also hop to a different part of the spectrum when needed.

Although the potential of cognitive radio networking appears promising, it entails several challenges that are not present in traditional wireless networks. One of the biggest challenges facing cognitive radio networking is how to find the set of *"available channels"*. Saying that a particular frequency channel is available for an SU means that no PU transmission that may be harmed by the SU transmission is currently active on that particular channel. Monitoring PU activity is usually referred to as *spectrum sensing* in the context of cognitive radio networking. The challenging issue in spectrum sensing is for an SU to conclude that there is an activity, i.e, transmission, on a particular channel and whether this activity corresponds to PUs and not to some other SUs. A more detailed review of spectrum sensing techniques and challenges are provided in Chapter 2.

Another big challenge in cognitive radio networks is spectrum sharing. Spectrum sharing defines the set of rules and strategies that regulate the behavior of secondary users regarding spectrum mobility, allocation, and access. In general, spectrum sharing architectures are classified into two categories: centralized and distributed [1]. For the centralized case, a spectrum management entity controls both spectrum allocation and spectrum access [4, 5, 6, 7]. In a distributed architecture, on the other hand, each SU is responsible for the channel allocation and access decisions. The SU may make its decisions based on its local observation of the network and spectrum status or by cooperating with other SUs to have a more global observation [8, 9, 10, 11].

1.1.1 Cognitive Radio Wireless Mesh Networks (CR-WMNs)

A wireless mesh network is a communication network that consists of a number of wireless nodes organized in a mesh topology. These wireless nodes are usually classified into three major categories based on their roles in the network: mesh routers (MRs), mesh clients (MCs), and gateway routers. Gateway routers connect the wireless mesh network to a backbone network, or the Internet. Each mesh router, on the other hand, manages a number of mesh clients in its cell and connects them with the backbone network over multiple hops of mesh routers, and eventually through gateway routers. Lastly, mesh clients are end-users' equipment, like desktop computers, laptops, cell-phones etc., used to connect the user to the mesh network. Figure 1.1 shows a general architecture of a wireless mesh network. Three network architectures are known for wireless mesh networks (WMNs) [12]. The first one is called *Infrastructure WMN* in which mesh routers form an infrastructure for mesh clients, and we assume this architecture in this dissertation. The second architecture is referred to as *clients WMN* in which no mesh routers are needed. In this architecture, mesh clients form the actual network to perform routing and network reconfiguration. Having the two previous architectures combined in the same network forms the third architecture, which is referred to as *Hybrid WMN*.

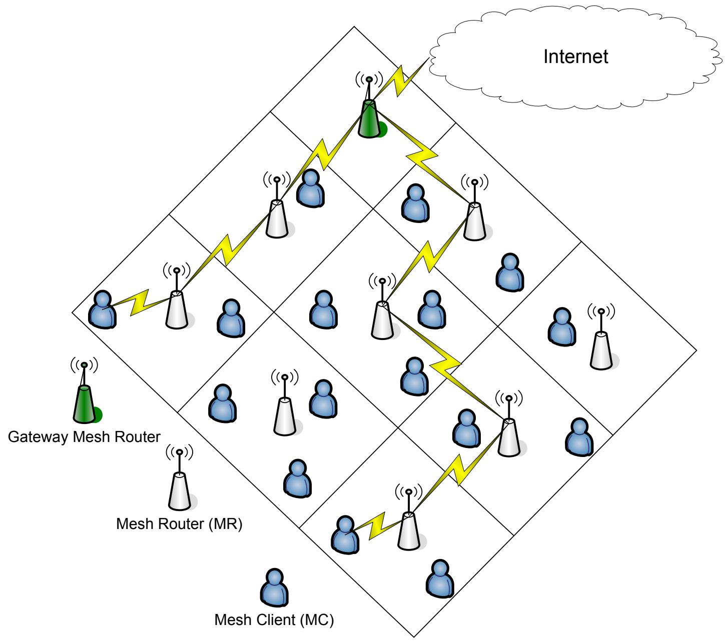


Figure 1.1: A general architecture of a wireless mesh network.

A cognitive radio mesh network is a WMN that deploys cognitive radios (for both routers and clients) and relies on opportunistic and dynamic spectrum access for its operation [13, 14, 15]. In addition to the fundamental motivations of increasing spectrum utilization and overcoming spectrum scarcity, cognitive radio mesh networks were motivated by a number of potential applications, like:

1. *Alleviating congestion in traditional WMNs*: exploiting cognitive radios allows mesh routers to search for available channels in the primary band (i.e., the licensed band) so they can reduce the congestion on the operational band of the WMN (usually a sub-band of the 2.4 GHz ISM band [1]) by shifting the mesh clients they serve to those available channels [14, 16].

2. *Increasing network coverage*: in some situations, mesh clients may demand some QoS guarantees regarding SINR (signal to interference and noise ratio) and BER (Bit-Error-Rate). To achieve such guarantees, mesh routers and clients need to restrict their transmission power levels so that the interference they cause at the location of other mesh

[1] The ISM band is a radio band reserved for industrial, scientific, and medical applications.

clients in neighboring cells stays within a pre-calculated threshold that insures the required SINR. However, restricting the transmission power means restricting the network coverage. Exploiting cognitive radios allows mesh routers to heal this problem by extending their coverage on any available channels in the primary, i.e., licensed, band [17].

3. *Integration of heterogeneous wireless access networks*: different heterogeneous wireless access networks currently exist, like wireless personal area networks (WPAN), wireless local area networks (WLAN), wireless metropolitan area networks (WMAN), and wireless regional area networks (WRAN). As spectrum reuse (provided that reliable communication is maintained for licensed systems) is becoming a crucial need to increase spectrum utilization and overcome spectrum scarcity, the merging of the aforementioned frequency separated access networks is inevitable. Recent research initiatives suggest the integration of these heterogeneous access networks into one cognitive radio mesh network using the ability of cognitive radios to adapt to different transmission/reception parameters like power, frequency, modulation, and medium access [18, 19]. For the mesh client side (especially mobile clients), deploying cognitive radios clients them to work with different wireless access networks, allowing seamless mobility. Therefore, a mobile mesh client can observe the performance (in terms of throughput, service availability, and reliability) of different coexisting wireless access networks and select the network that best fits its requirements [20]. Mesh routers, on the other hand, can also exploit cognitive radios to provide customized service to different mesh clients based on their requirements and interface standards.

The development of efficient protocols and algorithms that can boost the performance of cognitive radio mesh networks is the goal of this dissertation. In the following section, we layout the research issues addressed in this dissertation.

1.2 Research Issues

Cognitive radio networking entails a vast domain of research issues. The most critical among which is spectrum sensing, and it has been the focus of mainstream research in this area (see Section 2.1 for more details). Spectrum management and sharing is another important issue that has received considerable attention by the research community [1]. In this dissertation, we address three categories of networking research issues in cognitive radio wireless mesh networks that are explained the following subsections.

1.2.1 The communication coordination function and spectrum allocation

Although the technology of cognitive radios gives SUs more flexibility and adaptability (by allowing them to change their communication parameters dynamically), it makes the coordination of the communication process much more complicated. This complexity arises from the fact that SUs might be operating on different frequency channels at different times. This requires the communicating pair of SUs to negotiate their channel availability and decide on one channel for communication. But, the negotiation itself must take place over a common channel that is known to the communicating pair a priori; this channel is usually referred to as the *common control channel (CCC)*. At the network level, this *CCC* has to be common network wide to guarantee network operation. However, relying on a CCC has several drawbacks, like:

(1) Depending on the network size (area and number of nodes), the number and distribution of PUs, and the pattern of primary channels usage, the probability of having a CCC might be very low [8].

(2) From a security point of view, a denial of service (DoS) attack that jams the CCC will break the operation of the entire network.

(3) Sharing one control channel between all SUs might lead to congestion on this channel which will consequently cause performance degradation for the overall network.

These drawbacks motivated the research community to come up with alternative solutions to the CCC approach (we review some of these approaches in Chapter 2). A recent paradigm that assigns the task of spectrum sensing to a dedicated sensor network infrastructure (see Chapter 2 for details) frees SUs from the burden of spectrum sensing, and consequently from the need for a CCC to coordinate the sensing process. In effort to completely alleviate the need for a CCC, we propose in Chapter 3 a channel allocation mechanism that does not require a CCC to coordinate the communication process.

1.2.2 Performance degradation due to channel heterogeneity

Depending on the activity of the PUs in their vicinity, different SUs may observe different channel availabilities. This heterogeneity in channel availability leads to the following problems that affect all modes of communication, i.e., unicast, multicast, and broadcast:

1. *Broadcast deformation*: when an SU has neighbors that do not (all) share a common channel with this SU, it cannot broadcast a data unit to all neighboring SUs in one transmission. Therefore, a broadcast might become a number of multicast transmissions, or in the worst case a number of unicast transmissions. This leads to excessive reduction in network capacity and significant increase in the end-to-end delay especially in multicast applications.

2. *Switching delay*: another source of capacity wastage and delay increase is channel switching [21]. Assume that SU i receives from SU j and forwards the data to SU k. If i cannot find a common channel with j and k together, then it has to use two different channels for transmission over the two links $j \rightarrow i$ and $i \rightarrow k$. Depending on how separated the two channels are, the switching delay could be significant. The problem worsens when an SU has to receive from and/or transmit to multiple SUs.

In traditional multichannel wireless networks, the use of multiple radio interfaces was widely adopted as a solution to make full use of the capacity provided by the multiple channels and to avoid the switching delay problem [22]. However, it is usually the case in traditional multichannel wireless networks that the same set of channels is available to all nodes in the network, which is not the case in cognitive radio networks. Therefore, new solutions are needed for cognitive radio networks.

1.2.3 Dynamic spectrum availability

As spectrum availability is dynamic, in terms of both the set of available bands and the distribution of availability/outage time of different bands, a great challenge is imposed on the design of networking protocols and algorithms for cognitive radio networks. For example, for a routing protocol designed for a wireless cognitive radio mesh network to be efficient, it must have a reconfiguration capability that ensures fast response, i.e, re-routing, to any channel failure, i.e., channel unavailability, along a particular route. This fast response is important to minimize the potential network outage (disconnectivity) time.

1.3 Contributions of Our Research

In this dissertation, we study three research problems in cognitive radio wireless mesh networks (CR-WMNs). First, we study the problem of spectrum allocation with QoS guarantees in CR-WMNs assuming that both mesh clients (MCs) and mesh routers (MRs) employ cognitive radios, and that they rely on opportunistic spectrum access for their operation. Second, we study the problem of multicast scheduling in CR-WMNs. Our objective in this study is to propose a scheduling approach that can overcome the degradation in multicast throughput due to the channel heterogeneity problem. Third, we study the effect of channel switching on the performance of multicast routing in CR-WMNs. The contributions of this research are the following:

1. *Receiver-Based Channel Allocation* [17]: we propose a receiver-based channel allocation strategy that has the following advantages over traditional strategies.

   - It simplifies the communication coordination function in a CR-WMN by alleviating the need for a CCC.
   - It increases the service capacity, i.e., number of served MCs, in a CR-WMN.

   We formulate the proposed allocation strategy, as well as two other traditional strategies, as integer linear programs to obtain optimal solutions. We also prove the NP-hardness of the channel allocation problem in cognitive radio mesh networks. A heuristic algorithm with polynomial runtime is also proposed to obtain a suboptimal solution at a reduced complexity. Numerical results were obtained to evaluate the efficiency of the proposed heuristic solution.

2. *Assisted Multicast Scheduling* [23]: we propose a receiver-assisted multicast scheduling algorithm to reduce the effect of the channel heterogeneity problem on the multicast process. We also exploit network coding to further enhance the performance of the proposed scheduling algorithm. Integer linear program (ILP) formulations and heuristic algorithms were proposed for the studied problem. Furthermore, we propose solutions to build and maintain collision free schedules across different cells in the CR-WMN. Numerical results show that significant increase in multicast throughput can be achieved using the proposed assistance mechanism.

3. *On-demand Multicast routing* [24]: we study the effect of channel switching on the multicast routing performance. We also propose an on-demand multicast routing algorithm based on dynamic programming. The algorithm tries to jointly find a route to the gateway and allocate channels along that route to minimize the end-to-end delay. The gain of the proposed approach is validated through simulation.

1.4 Organization

The rest of this dissertation is organized as follows. We review related work in chapter 2. A receiver based channel allocation strategy for cognitive radio mesh networks is proposed in Chapter 3. In Chapter 4, we propose a multicast scheduling mechanism that reduces the effect of channel heterogeneity on the multicast throughput through receiver assistance and network coding. An on-demand multicast routing algorithm that takes into consideration the heterogeneity in channel availability and its effect on the end-to-end delay is presented in Chapter 5. We conclude this dissertation and layout our future work directions in Chapter 6.

CHAPTER 2. RELATED WORK

In this chapter, we review the literature on some related research issues. Although we do not address the problem spectrum sensing in this dissertation, we review the work that has been done in this area so we can comment later on the compatibility of our solutions with each of the existing spectrum sensing techniques. Then, we summarize the solutions that have been proposed over the past few years to overcome the challenging drawbacks of the common control channel (CCC) approach. After that, we review some very recent research efforts addressing the multicast problem in cognitive radio networks. Then, we give a brief introduction to network coding and review some studies that used this technique in cognitive radio networks. Finally, we review some related work on the routing problem in cognitive radio networks.

2.1 Spectrum Sensing

Two main tracks shape the research efforts undertaken by the wireless communication research community over the past decade to reach to an efficient spectrum sensing mechanism, namely cooperative and non-cooperative sensing. In non-cooperative sensing [25, 26, 27], each SU relies on its sole effort to detect the presence of a PU transmission on a particular spectrum band. Two methods were proposed in literature to detect the presence of a PU transmission without any form of cooperation between SUs, and those are *energy detection* [25, 26, 28] and *signal feature detection* [29, 27, 30]. Using the energy detection approach, an SU concludes that a neighboring PU is receiving on a particular spectrum band if a weak PU signal (the signal's energy is above certain threshold) is detected at the SU. This approach suffers from two major disadvantages:

1. *Cumulative noise*: this approach does not distinguish between the actual nearby trans-

Table 2.1: A summary of the characteristics of different spectrum sensing paradigms.

Sensing Paradigm	SUs need a CCC	Why CCC is needed?	Accuracy
Non-Cooperative	Yes	Synchronizing silence periods	Low
Cooperative	Yes	Exchanging sensing information	Medium
Sensor network-aided	No	-	High

mission of a PU and the case of accumulated noise that exceeds the detection threshold. Therefore, false-positives may be encountered by SUs under such approach which will limit spectrum accessibility, especially is noisy environments.

2. *Silence synchronization*: another major problem of energy detection sensing is the fact that an SU cannot distinguish between PUs transmissions and fellow SUs transmissions. Therefore, an SU may avoid accessing a particular spectrum band because of the current activity of other SUs on that band, and not because the licensed users (PUs) are actually using it. To overcome this problem, researchers proposed that SUs must synchronize their sensing activity, in a so called silence period, so that the case of detecting SU transmissions instead of PU transmissions is avoided. However, this solution poses another challenge, which is how to synchronize SUs. The widely adopted approach to tackle this challenge is to exploit a control channel that is common, i.e, available, to all SUs which is usually referred to the *common control channel* (CCC). Finding a CCC is a very challenging issue that has received a considerable attention over the past few years. We will address the CCC issue in more detail later in this chapter.

The second method used in non-cooperative sensing is *signal feature detection*. Using this method, an SU looks for some feature in the received signal to confirm that it is coming from a PU. These features are usually modulation-dependent. Most modulated signals have some built-in features, like for example the periodicity of signals transmitted over sine-wave carriers, that make them distinguishable from noise. Therefore, this method is more resistant to the *cumulative noise* problem from which the energy detection method suffers. However, feature detection sensing is computationally complex and requires longer detection time.

On other hand, cooperative sensing [31, 32, 33, 34, 35, 36] allows SUs to cooperate and consolidate their spectrum sensing efforts in order to reach a more accurate conclusion about spectrum availability. This approach of spectrum sensing comes as a solution to a major drawback of non-cooperative sensing, that is the effect of the hidden node problem on the sensing outcome. If the SU which is currently sensing the spectrum, non-cooperatively, is hidden from the PU transmitter, it will not be able to detect the PU signal. Therefore, the SU will conclude that the corresponding spectrum band is vacant, and start using it and consequently harming the PU receiver. The same scenario may happen due to shadowing. Such scenarios can be avoided if SUs cooperate with each other to estimate spectrum status instead of independently doing this estimation on their own. Cooperative sensing can be either distributed [34] or centralized [33, 35]. In the distributed implementation, SUs exchange their personal sensing outcomes over a CCC and then use a certain estimation model to map all received outcomes to a final decision about the status of the corresponding spectrum band. In the centralized implementation, on the other hand, all personal sensing outcomes are sent, over a CCC, to a centralized entity where the decision about spectrum status is made and then sent back, again over the CCC, to SUs.

Recently, a new spectrum sensing paradigm based on wireless sensor networks has emerged, and it is usually referred to as sensor-network-aided spectrum sensing [37, 38, 39]. Under this paradigm, an infrastructure sensor network dedicated for spectrum sensing is deployed in a field to allow coexistence between licensed and unlicensed users. SENDORA, SEnsor Network for Dynamic and cOgnitive Radio Access [40], is a research project started in 2008 by a number of research institutes in Europe that adopts the sensor network paradigm to obtain high spectrum sensing accuracy, and therefore enable coexistence between licensed and unlicensed users. SUs can obtain information about spectrum availability by querying the infrastructure sensor network. This paradigm has the following advantages:

1. *Simplicity for SUs*: under this paradigm, SUs are free from the burden of spectrum sensing. Therefore, they do not need to synchronize their silence periods as required by the non-cooperative sensing paradigm, or hunt for a common control channel as required

by cooperative sensing.

2. *Higher sensing accuracy*: as the sensor network is deployed as an infrastructure network dedicated for spectrum sensing, higher sensing accuracy can be achieved by carefully designing the sensor network (in terms of density and deployment). This basically means better service for SUs and better protection for PUs.

3. *Promising potential*: this sensing paradigm paves the way for a new generation of wireless technology and communication paradigms that can highly enhance spectrum utilization and provide a new opportunistic class of services at a fairly low cost for unlicensed users, while at the same time reducing the service cost for licensed users. This can be achieved using spectrum leasing [41]. A PU may lease the spectrum it uses in its vicinity, i.e, allowing interference in its vicinity on the leased spectrum channel, to SUs through the infrastructure sensor network when it (the PU) is not in need for the spectrum.

Table 2.1 summarizes the major differences between the three spectrum sensing paradigms discussed in this section. Current research efforts that propose architectures and frameworks for cognitive radio mesh networks rely on cooperative spectrum sensing to identify spectrum holes [42, 14]. Mesh clients in a cell sense the spectrum and send their sensing results to the mesh router managing that cell, which may cooperate with other mesh routers to reach an accurate estimate of the spectrum status. Other research efforts rely on non-cooperative sensing such that each SU builds its list of available channels based on the outcome of its own spectrum sensing [43]. The solutions proposed in this dissertation are independent of the spectrum sensing paradigm as we avoid the cross-layer design [17, 23].

2.2 The Common Control Channel (CCC) Problem

A number of alternatives to the fixed CCC approach were proposed in literature. We review some of these alternative solutions next.

Local control channels instead of a single control channel common to all SUs [44], Zhao et al. [8] proposed the use of local control channels (LCCs) each of which is common to only a group of SUs. Using this approach, SUs at group boundaries, i.e., those that have neighbors in two different groups, might have to use (listen to and transmit on) more than one control channel. Although this approach is better than the CCC approach, it has its own drawbacks. First, the jamming problem is not alleviated although the scale of its effect is reduced to a group-level rather than a network-level. Second, SUs at group boundaries need to be either equipped with multiple transceivers or keep switching a single transceiver to listen to multiple LCCs, as well as the data channel. This will result in an increase in inter-group communication delay and degradation in network throughput.

Selective broadcasting another alternative solution is to broadcast control information either over all available channels [45] or over a small subset of channels which covers all the neighbors of a node [46]. In [46], each node transmits the control information on a selected group of channels instead of a single control channel and this is why the approach is called selective broadcasting. Similar to the previous solution, a node might have to listen to more than one control channel. This requires the channel activity operations (listening/transmitting) to be synchronized in order for a communicating pair of nodes to successfully exchange control information.

Common Frequency Hopping the third alternative is to have all SUs share a common frequency hopping sequence which they use for spectrum sensing and data transmission [47, 48]. This common sequence will guarantee that a pair of neighboring SUs that both observe a particular licensed channel to be available will be tuned to that channel at the same time. Therefore, communication can take place without coordination between transmitters and receivers. This approach faces a number of challenging issues. First, there is the need to maintain the synchronization of the hopping process for such an approach to operate effectively. Second, switching the radio transceiver to a different frequency band might take a non-negligible amount of time, referred to as the switching latency, that is usually directly proportional to the separation between the current and the new frequencies.

2.3 Multicasting in Cognitive Radio Networks

Fundamental operational issues, like spectrum sensing and management, have been the focus of mainstream research in the area of cognitive radio networking for many years. Therefore, there was little research on upper-layer issues like routing and group communication. Recently, a number of studies have come to focus on multicasting in CRNs.

In [49], a cross-layer optimization approach was proposed to enable video multicasting in CRNs. The study aims at optimizing the overall received video quality, while achieving fairness among multicast users and avoiding interference with licensed users. This study takes into consideration different factors like video coding, video rate control, spectrum sensing, spectrum access, modulation, and multicast scheduling.

Subcarrier assignment and power allocation to multicast groups of secondary users, under the constraint of tolerable interference to primary users, was studied in [50] for OFDM-based cognitive radio multicast network. This work considers a single-cell wireless system in which a single base station (BS) serves both primary and secondary users. Subcarriers in the available spectrum that are not used for transmitting to PUs are exploited, by the BS, to serve SUs. As subcarries may not necessarily be orthogonal, mutual interference between PUs and SUs to the transmission on adjacent subcarriers was also considered in the paper. The results of this can only be applied to scenarios in which the primary network manages the spectrum, through the BS, and determines what parts of the spectrum to sublease to SUs. Therefore, SUs cannot form their own network. Instead they have to be second-class users in an existing primary network.

One of the very first (and few) studies that tackle the multicast-tree construction in CRNs is [43]. The proposed approach tries to minimize the number of forwarding nodes (nodes that are non-receiver members of the multicast tree) to minimize the switching delay by adopting a tree structure routing rather than a mesh style. Switching delay is attributed to two sources at forwarder nodes; receiving data on one channel and forwarding it on another, or forwarding the packet to multiple nodes on different channels. This approach is not optimal as the channel assignment is made locally at each node without taking the total switching time over the tree into consideration.

The multicast capacity of multihop cognitive radio networks was studied in [51]. The paper studies a network model that consists of two ad hoc networks. The first is a primary (i.e., licensed) ad hoc networks (PaN), and the second one is a secondary (i.e., unlicensed) ad hoc network (SaN). The two networks overlap in both space and spectrum. Primary and secondary nodes are distributed in a unit square region according to a poisson point process with intensities n and m respectively. The paper assumes the existence of multicast traffic in both the PaN (n_s sessions exist) and the SaN (m_s sessions exist) networks. Sizes and actual members of the multicast sessions in both networks are selected randomly. The paper tries to answer the following question: *given that the PaN network adopts a multicast protocol that can achieve the optimal throughput (with the SaN out of the picture), what is the maximum achievable multicast throughput for the SaN?* Under the system model described earlier, and the assumption that both networks operate a TDMA-based medium access scheme with equal slot times, the paper concludes that both networks can achieve their asymptotic capacities as if they were operating disjointly through careful design of the multicast schemes.

2.4 Network Coding in Cognitive Radio Networks

Network coding (NC) [52, 53] has emerged as a very promising technique to enhance multicast throughput [54, 55] and provide protection and survivability [56, 57, 58] in wireless networks. This technique allows nodes in a network to combine packets instead of just forwarding them unchanged. Three variants of network coding have been proposed in literature, namely digital network coding (DNC) [52], physical-layer network coding (PNC) [59], and analog network coding (ANC). DNC requires packet combining to be done at layer 2 or above, i.e., both packets must be correctly and independently received before combining. In PNC. packets, or more precisely the electromagnetic waves of those packets, are combined at the physical layer. Then, the combined signal is mapped into an interpretable output signal that is then forwarded to the final destinations. PNC works for 2-way relay network, and requires symbol-level, carrier-frequency, and carrier-phase synchronization between the two combined signals. ANC was proposed to overcome this synchronization problem in PNC, and it in fact relies on the lack of synchronization for its operation, while keeping the coding operation at the signal level [60].

Very few studies have proposed applying NC to CRNs. Recently, a study was conducted in [61] to exploit network coding to increase the throughput of SUs in a CRN and decrease the amount of incurred interference and consumed energy. In a more recent study [62], the capacity of a cognitive radio relay network was investigated. Network coding is utilized to enable a cognitive system (pairs of SUs and sinks) to exchange information through a cooperative relay network that a primary system (pairs of PUs and sinks) is using to relay its own traffic. The effect of the interference of the cognitive system on the capacity of the primary system is studied. Moreover, the maximum achievable capacity for the cognitive system has also been studied. This work has shown that using network coding to make primary resources available for a cognitive system through cooperative relaying is a viable approach that achieves spectrum efficiency at network throughput. It is to be noted that these two studies use digital network coding (DNC).

Analog network coding (ANC) was used in [63] to enhance the throughput of Inter-cluster communication in wireless cognitive radio mesh networks. Gateway nodes (those that are connecting clusters) exploit ANC to relay data between two adjacent clusters with minimum time, while at the same time control their transmission powers such that no harmful interference is caused to PUs. In another study, [64], network coding was used as a means to enhance the performance (throughput) of a distributed virtual control channel proposed to overcome the common control channel problem. The basic idea is allow SUs to exchange control information whenever they meet on a particular channel, exploiting network coding for better dissemination performance.

2.5 Routing in Cognitive Radio Networks

The current literature work in cognitive radio networks (CRNs) is mostly focused on efficient spectrum utilization, and few studies consider the effect of heterogeneity in channel availability on multicast routing performance. In [65], a cross-layer optimization approach for video multicast on heterogeneous channels was proposed. The study aims at optimizing the overall received video quality, while achieving fairness among multicast users and avoiding interference with licensed users. The work in [66], on the other hand, proposes multihop multicast protocols for cognitive radio networks by employing techniques of network coding, power control and cooperative communication. In [67], the problem of constructing minimum energy multicast tree in CRNs is studied. The study considers the energy consumption at SUs due to spectrum sensing and data transmission.

An algorithm for routing and channel allocation in CRNs based on a layered graph model was proposed in [68]. The algorithm assumes multiple interfaces per node, and aims at maximizing network connectivity. It also tries to diversify the channel in order to minimize the interference between adjacent links.

Given the dynamism of channel availability in cognitive radio networks, route maintenance comes into the picture as an important routing metric. Minimum maintenance cost routing in cognitive radio networks was studied in [69]. The problem was formulated as an integer program optimization, for which a heuristic alternative was also proposed.

As cognitive radio networks can operate on a very wide spectrum of channels, there is a need for routing strategies that take the extra, potentially significant, latency and throughput degradation caused by switching the radio interface between frequency channels. We propose such a strategy in Chapter 5.

CHAPTER 3. CHANNEL ALLOCATION FOR MAXIMIZED COVERAGE AND SIMPLIFIED COORDINATION

In this chapter, we study the channel allocation problem in cognitive radio wireless mesh networks (CR-WMNs). We aim at finding an allocation strategy that guarantees quality of service (in terms of link reliability), maximizes network coverage, and alleviates the need for a common control channel to coordinate the communication process. The allocation of a particular channel to a mesh client (MC) is considered feasible if the MC can establish connectivity with the backbone network in both upstream and downstream directions, and has the SINR (signal to interference and noise ratio) of the uplink and the downlink with its parent mesh router (MR) within a predetermined threshold. A receiver-based channel allocation model that achieves the aforementioned objectives is proposed in this paper. We prove that the channel assignment problem using the proposed receiver-based approach is NP-complete. We then formulate a mixed integer linear program (MILP), of the proposed model, and compare its performance to that of two other baseline models, namely, transmitter-based and all-tunable channel allocation strategies. The results prove the superiority of the proposed model. Due to the hardness of the problem, we also develop a heuristic algorithm, which is shown to be an accurate algorithm.

The problem of channel allocation in CR-WMNs has been addressed in a number of studies over the past few years. In [44], a cluster-based approach was proposed such that the network is clustered into 1-hop clusters based on channel availability. Nodes that belong to the same cluster use the same control channel. Inter-cluster communication takes place through gateway nodes that operate on multiple control channels. Therefore, the channel allocation problem was mainly the allocation of control channels. This LCC based approach does not restrict the users to use specific data channels, but rather allows them to negotiate their data channels on the control channel. Therefore, no inter-cell interference guarantees were obtained.

In [70], the problem of downlink channel assignment and power control was studied. The objective was to assign channels to downlinks such that the interference at PU locations is bounded and the downlink channel reliability is above a predefined threshold. This work considers the downlink case only and does not study the uplink case and the overall network connectivity. Our work in this chapter, on the other hand, considers all the three aspects together, namely, the downlink, the uplink, and the network connectivity.

The problem of channel selection in multihop cognitive mesh networks was studied in [71]. The main objective was to select a channel that a node can transmit on such that the interference temperature[1] within the transmission range of the node does not exceed a predefined threshold. Fixed and adaptive power control strategies were proposed for this purpose.

The rest of this chapter is organized as follows. In Section 3.1, we present the network model and layout the assumptions of this work. The receiver-based channel allocation strategy is presented in Section 3.2. In Section 3.3, we study the complexity of the receiver-based channel allocation problem and propose a mixed integer linear program (MILP) formulations for different channel allocation strategies. In Section 3.4, we propose a heuristic algorithm for the receiver-based channel allocation problem in wireless cognitive mesh networks. The performance of the proposed heuristic algorithm is evaluated in Section 3.6. We also evaluate the optimal performance of the proposed receiver-based channel allocation strategy versus other possible allocation strategies in Section 3.6.

3.1 System Model

In this section, we present the system model and assumptions and state the objectives of this chapter.

[1] A new metric proposed by FCC that accounts for the cumulative radio frequency energy from transmissions and sets a maximum limit on their aggregate level.

3.1.1 Assumptions

The general network structure consists of a number of MRs, some of them are directly connected to a backbone network, each of which manages all the MCs in its cell. This structure has the following properties:

- Each MC is associated with exactly one MR, and communication within the cell takes place in one hop.

- All MCs and MRs are equipped with cognitive radios, and they communicate with each other through wireless communication over the unused licensed channels to reach the backbone network.

- We assume that a subset of MRs, that we call *gateways*, are directly connected to the backbone network. Therefore, each non-gateway MR should be able to reach at least one of the gateways in multiple hops of MRs in order to establish connectivity with the backbone network. From now on, we use the word *gateway* to refer to an MR that is directly connected to the backbone network, and use the abbreviation *MR* to refer to a mesh router regardless of whether it is a gateway or not, and use the word *node* to refer to an SU (MR or MC).

- For each served MC, the reliability of its uplink ($MC \rightarrow MR$) and that of its downlink ($MR \rightarrow MC$) must meet a given QoS requirement (a threshold reliability).

Throughout this chapter, we assume the following:

- The channel availability at a node (MC or MR) is quasi-static, i.e., the channel status does not change in a short period of time. Therefore, this work is more suited to spatial spectrum underutilization than temporal underutilization. However, it can still be used for the case of temporal spectrum underutilization if the PU activity (occupying and vacating channels) is not very dynamic.

- A simple path loss model for channel attenuation.

- The CR-WMN relies on a spectrum management and sensing entity (SSME) to identify the set of available channels at each node. This entity shall use any of the sensing approaches proposed in literature, also reviewed in Section 2.1.

3.1.2 Objective

Our objective is to evaluate the performance of different channel allocation strategies for CR-WMNs. For any allocation strategy to be feasible, the following two conditions must be satisfied for all served MCs.

(1) A path from the MC to at least one gateway must exist; we call this the *upstream connectivity constraint*. Also, a path from at least one gateway to the MC must exist; we call this the *downstream connectivity constraint*. The two paths may be disjoint, also the upstream and downstream gateways may be different.

(2) Potential interference caused by intra-cell communication from cells other than the parent cell of an MC must be bounded to achieve a predetermined SINR to guarantee a BER (bit error rate) QoS requirement.

Therefore, we aim at finding the best channel allocation strategy that satisfies the above two conditions for the maximum number of MCs.

3.2 Receiver-Based Channel Allocation

Based on the joint temporal and spatial distribution of the availability of the licensed spectrum, different SUs might observe different sets of available channels. Therefore, four modes of operation can be defined for each node's transceiver:

1. *Tunable Transmitter - Tunable Receiver (TT-TR)*: an SU can transmit/receive on any of the available channels.
2. *Tunable Transmitter - Fixed Receiver (TT-FR)*: an SU can transmit on any of the available channels, but receives on a fixed channel.

3. *Fixed Transmitter - Tunable Receiver (FT-TR)*: an SU can receive on any of the available channels, but transmits on a fixed channel.

4. *Fixed Transmitter - Fixed Receiver (FT-FR)*: an SU transmits/receives on a fixed channel.

TT-TR is the most commonly assumed communication paradigm in multihop cognitive radio networks. It allows an SU to use any of its available channels for transmission and/or reception. Therefore, the channel allocation problem under this paradigm will be to assign channels to links. This means that a node might use different channels for its incoming and outgoing communication links with its neighbors. The drawback of this paradigm, which is shared with the *FT-TR* paradigm, is that a common control channel (CCC) is needed for channel negotiation. In other words, these two communication paradigms cannot be used without a CCC because the transmitter needs to inform the receiver about its intention to transmit so that the receiver can tune to the transmitter's channel. Because of the problems of the CCC approach discussed earlier, and the fact that the probability that a CCC exists could be low [8], it is a necessity to devise a new communication paradigm in which the requirement of a CCC is avoided. In this chapter, we propose a channel allocation strategy based on the *TT-FR* paradigm, we call this strategy a *receiver-based channel allocation (RBA)*. Based on this allocation strategy, each node (MC or MR) is allocated a fixed channel to receive on, but it is allowed to transmit on different channels. Therefore, if each node knows the channels allocated to its neighbors, no channel negotiation is needed, and consequently no CCC is needed as well for this purpose. *To be specific, if node A wants to communicate with node B, where B is assigned channel f_B, then node A must have channel f_B among its list of available channels. If so, node A tunes its transceiver to f_B and initiates communication with B according to the MAC mechanism used. Then, communication takes place on channel f_B.* We do not consider the $FT-FR$ mode in this study because it is a special case of the $TT-FR$ that has advantage of not requiring a CCC, but the disadvantage of limited connectivity.

3.2.1 Issues and challenges

In this subsection, we would like to emphasize on some issues and challenges related to the proposed RBA approach.

Degree of connectivity it is expected that the assignment of one channel for each node to receive on will result in a decreased level of connectivity. One might think that the *TT-TR* approach will result in the highest degree of connectivity, because transmission and reception are allowed on any channel, which intuitively implies that the number of served MCs will be higher compared to *TT-FR*. However, this is not necessarily true because of the following:

- The *TT-TR* approach requires the existence of a CCC, while the *TT-FR* approach does not, and the probability that the network is connected depends on the probability that a CCC exists.
- The channel used as a CCC cannot be used for data communication. This gives the *TT-FR* one more channel to use for data communication.
- The effect on connectivity depends on the channel availability distribution at a node and its neighbors. It is more likely for a channel available at a particular node to be available at its neighbors, which lowers the likelihood that a node gets disconnected from its neighbors when it is assigned a fixed channel to receive on.

Deafness Problem the deafness problem is recognized in wireless networks with directional antennas [72]. Deafness is caused when node A wants to communicate with node B while B is currently communicating with node C. Node A translates the absence of a reply from B (caused by the fact that B's antenna is tuned to the direction of node C) as a collision at B, and consequently backs-off [2]. The problem becomes worse if B has multiple packets to transmit, which will cause A to unnecessarily back-off several times. The same problem may occur under the proposed RBA approach. Let f_A, f_B, and f_C be the frequency channels

[2] This is based on CSMA/CA medium access.

assigned to nodes A, B, and C for reception respectively . Assume that node A wants to communicate with node B which is currently communicating with node C on channel f_C. Then, A will fail to reach B on f_B because B is currently tuned to f_C, and unnecessarily backs-off, i.e., node A, which results in the same deafness problem recognized in the directional antennas case. This problem is not present in the *TT-TR* and *FT-TR* approaches because the neighbors of the transmitter will know about the ongoing communication by overhearing the control information transmitted on the CCC to set up the communication session.

Multi-channel hidden node problem the last challenge is the hidden node problem. This problem is well known in wireless communication, however, it is a bit different using the *TT-FR* approach. In traditional single-channel wireless networks, under an IEEE 802.11 based MAC protocol, an RTS/CTS handshake between the transmitter and the receiver solves the hidden node problem. However, following the *TT-FR* mode, hidden nodes may exist on multiple channels. This means that the RTS/CTS handshake must be cloned on multiple channels, which results in significant delay if a single radio interface is used.

3.3 MILP formulations for the channel allocation problem

Before we give a formal definition of the receiver-based channel allocation problem in CR-WMNs, we present some notations and terminology.

- $\mathcal{B}$ is the set of non-gateway MRs .
- $\mathcal{G}$ is the set of gateways.
- $\mathcal{A}_i$ is the set of MCs that belong to the cell administrated by MR i. $\mathcal{A}$ is the set of all MCs in the network, i.e., $\mathcal{A} = \bigcup_{i\in\mathcal{B}\cup\mathcal{G}} \mathcal{A}_i$.
- $\mathcal{L}_i$ is the set of available (idle) channels at node i obtained from the infrastructure sensor network. And let $\mathcal{L}$ be the set of all available channels in the system such that $|\mathcal{L}| = K$. These channels are assumed to be orthogonal to each other. Moreover, $\mathcal{L}_j \subseteq \mathcal{L}_i \; \forall j \in \mathcal{A}_i$ because an MC cannot make use of a channel that is not available at its parent MR.

- P_i^k is the transmission power of node i on channel k such that $P_i^k \leq P_r^{max}$ $\forall i \in \mathcal{B} \cup \mathcal{G}$, and $P_i^k \leq P_c^{max}$ $\forall i \in \mathcal{A}$. P_r^{max} and P_c^{max} are the maximum transmission powers of an MR and an MC respectively.

- Ψ_{ij}^k is the channel power gain from i to j on channel k.

- ζ_{ij}^k is the maximum amount of interference that the cell managed by MR i may produce at the location of node j on channel k.

$$\zeta_{ij}^k = \max\{\max_{w \in \mathcal{A}_i} P_w^k \Psi_{wj}^k, P_i^k \Psi_{ij}^k\} \tag{3.1}$$

- ζ_{ij}^{max} is the maximum amount of interference that the cell managed by MR i produces at the location of node j at maximum transmission power, i.e.,

$$\zeta_{ij}^{max} = \max\{\max_{w \in \mathcal{A}_i} \max_{k \in \mathcal{L}_w} P_c^{max} \Psi_{wj}^k, \max_{k \in \mathcal{L}_i} P_r^{max} \Psi_{ij}^k\} \tag{3.2}$$

- N_0 is the channel noise power, and it is assumed to be the same at all locations on all channels.

- γ is the minimum SINR value required to guarantee a certain BER at a node (reliability threshold).

- c_j^k is a binary variable that is set to 1 if channel k is assigned to node j, and 0 otherwise.

3.3.1 Receiver-based channel allocation (RBA) problem

The receiver-based channel allocation (RBA) problem in wireless cognitive mesh networks is defined as follows:

Definition 3.3.1. *RBA problem: given a wireless cognitive mesh network of $\mathcal{G}$ gateway MRs, $\mathcal{B}$ non-gateway MRs, and $\mathcal{A}_i$ MCs managed by MR i for all $i \in \mathcal{B} \cup \mathcal{G}$. Also, for all $j \in \mathcal{B} \cup \mathcal{G} \cup \mathcal{A}$, the geographic location of j and its channel availability $\mathcal{L}_j$ are given. Find a $TT-FR$ channel allocation that maximizes the number of served MCs such that for each served MC, the following conditions are satisfied:*

1. *A path from each MC (through its parent MR) to at least one MR in $\mathcal{G}$ exists.*

2. *A path from at least one MR in $\mathcal{G}$ to each MC (through its parent MR) exists.*

3. *The SINR of the uplinks ($MC \rightarrow MR$) and the downlinks ($MR \rightarrow MC$) is at least γ.*

Note that the upstream and downstream paths for an MC must go through its parent MR. Therefore, the RBA problem be decomposed into two subproblems: (1) *channel allocation to MRs such that the upstream/downstream connectivity constrain is satisfied for MRs.* (2) *channel allocation to MCs such that reliable uplinks/downlinks with MRs are established for the maximum number of MCs.* The first subproblem can be represented as a network flow formulation as we show throughout this subsection. By adding few more constraints to jointly model the second subproblem, the whole RBA problem can then be formulated as an MILP.

To show the complexity of the RBA problem, let us just consider the upstream/downstream connectivity subproblem.

Definition 3.3.2. *Upstream/downstream connectivity problem (UDCP): given the network graph $G(\mathcal{B}\cup\mathcal{G}, E)$, where E is the set of connectivity edges between MRs, and the channel availability at each MR ($\mathcal{L}_i \forall i\in\mathcal{B}\cup\mathcal{G}$). Is there a receiver-based channel assignment that guarantees upstream and downstream connectivity for each non-gateway MR.*

Theorem 3.3.1. *The* UDCP *problem is NP-complete.*

Proof. See appendix B. □

It can also be shown that the optimization version of the UDCP problem (connectivity is established for the maximum number of MRs) is NP-hard.

Theorem 3.3.2. *The RBA problem is NP-hard.*

Proof. As UDCP is a subproblem of the RBA problem, and UDCP is NP-hard, then the RBA problem is at least as hard as the UDCP problem. □

Let us start with the network flow formulation for the first subproblem, i.e., upstream/downstream connectivity. Define a graph $G = (V, E \cup \overline{E})$ of a set of vertices $V = \mathcal{B} \cup \mathcal{G} \cup \{s, \overline{s}, d, \overline{d}\}$ and a set of edges $E \cup \overline{E}$. The vertices s and d represent a hypothetical source and hypothetical sink for the upstream flow respectively. On the other hand, $\overline{s}$ and $\bar{d}$ represent a hypothetical source and a hypothetical sink for the downstream flow respectively. E and $\overline{E}$ are the sets of upstream and downstream edges respectively. The set E is defined as follows:

- A directed edge $e = (s, j)$ exists for each vertex $j \in \mathcal{B} \cup \mathcal{G}$. The flow on such an edge is equal to the number of served MCs that belong to $\mathcal{A}_j$, i.e., $\sum_{i \in \mathcal{A}_j} \sum_{k \in \mathcal{L}_i} c_i^k$.
- A directed edge $e = (i, d)$ exists for each vertex $i \in \mathcal{G}$.
- A directed edge $e = (i, j)$ exists for any pair of MRs $i, j \in \mathcal{B} \cup \mathcal{G}$ if the following condition is satisfied for at least one channel $k \in \mathcal{L}_i \cap \mathcal{L}_j$,

$$\Psi_{ij}^k P_r^{max} \geq P_{th}, \tag{3.3}$$

  where P_{th} is a threshold received signal strength requirement to detect the transmission. The capacity of such an edge is given as $|\mathcal{A}| \cdot \sum_{k \in \mathcal{L}_i \cap \mathcal{L}_j} c_j^k$, i.e., the end-node j must be assigned a channel that belongs to the list of available channels at the start-node i in order for a flow upper-bounded by $|\mathcal{A}|$ to pass through the edge e.

On the other hand, the set $\overline{E}$ is defined as follows:

- A directed edge $e = (\overline{s}, j)$ exists for each vertex $j \in \mathcal{G}$.
- A directed edge $e = (i, \overline{d})$ exists for each vertex $i \in \mathcal{B} \cup \mathcal{G}$. The flow on such an edge is equal to the number of served MCs that belong to $\mathcal{A}_i$, i.e., $\sum_{j \in \mathcal{A}_i} \sum_{k \in \mathcal{L}_j} c_j^k$.
- A directed edge $e = (i, j)$ exists for any pair of MRs $i, j \in \mathcal{B} \cup \mathcal{G}$ if condition (3.3) is satisfied for at least one channel $k \in \mathcal{L}_i \cap \mathcal{L}_j$. The capacity of such an edge is given as $|\mathcal{A}| \cdot \sum_{k \in \mathcal{L}_i \cap \mathcal{L}_j} c_j^k$.

We define two flow commodities, one for the upstream flow (through the edges in E) and one for the downstream flow (through the edges on $\bar{E}$. Let f_{ij} denote the flow on edge $(i, j) \in E$, i.e. *upstream flow*, and g_{ij} denote the flow on edge $(i, j) \in \bar{E}$, i.e. *downstream flow*. The network flow representation is shown in Figure 3.1.

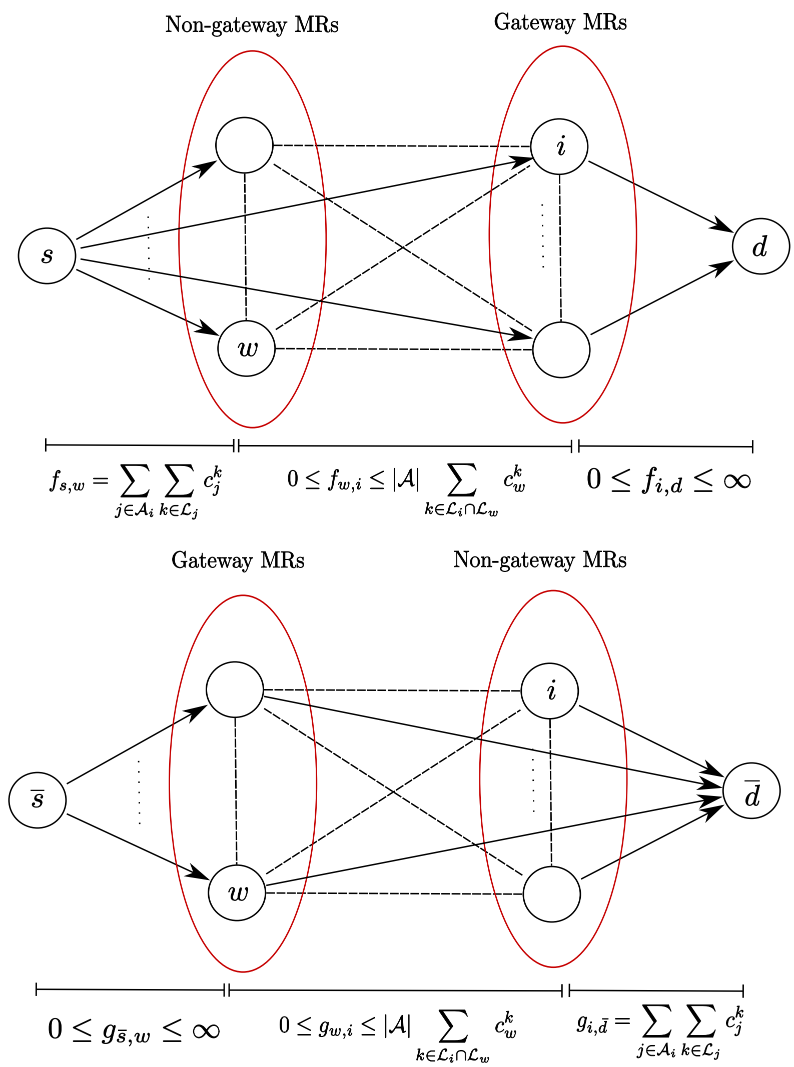


Figure 3.1: The network flow representation of the upstream (left) and the downstream (right) connectivity. Bounds on the flow on the three groups of edges (from source to MRs, between MRs, and from MRs to destination) are shown below the graph drawing.

Let us now consider the second subproblem we mentioned earlier, i.e., channel allocation to MCs. First of all, it is of no benefit to assign a channel to an MC j unless its parent MR, say i, is assigned one that is common between the two, otherwise, MC j will not be able to access its parent MR i which means that j cannot be served. Therefore,

$$\sum_{k \in \mathcal{L}_j} c_j^k \leq \sum_{k \in \mathcal{L}_i \cap \mathcal{L}_j} c_i^k, \ \forall \ i \in \mathcal{B} \cup \mathcal{G}, \ j \in \mathcal{A}_i$$

Then, the downlink (from an MR to an MC) reliability is achieved if the following inequality is satisfied:

$$\begin{aligned} &\Psi_{ij}^k P_i^k - \gamma \left(N_0 + \textstyle\sum_{m \in \mathcal{B} \cup \mathcal{G} \setminus \{i\}} \zeta_{mj}^k \right) \geq \\ &\gamma(c_j^k - 1) \left(N_0 + \textstyle\sum_{m \in \mathcal{B} \cup \mathcal{G} \setminus \{i\}} \zeta_{mj}^{max} \right), \\ &\forall i \in \mathcal{B} \cup \mathcal{G}, j \in \mathcal{A}_i, k \in \mathcal{L}_i \cap \mathcal{L}_j \end{aligned} \tag{3.4}$$

Note that if MC j is assigned channel k, then the right hand side is equal to 0 and P_i^k must be set to a value that satisfies the required SINR threshold γ at MC j. On the other hand, if MC j is not assigned channel k, i.e., $c_j^k = 0$, then the inequality is satisfied for any positive (≥ 0) value of P_i^k because the term between parentheses in the right-hand side is an upper bound on the term between parentheses in the left-hand side.

Similarly, the uplink (from an MC to an MR) reliability is achieved if the following inequality is satisfied:

$$\begin{aligned} &\Psi_{ji}^k P_j^k - \gamma \left(N_0 + \textstyle\sum_{m \in \mathcal{B} \cup \mathcal{G} \setminus \{i\}} \zeta_{mi}^k \right) \geq \\ &\gamma(c_i^k + \textstyle\sum_{w \in \mathcal{L}_j} c_j^w - 2) \left(N_0 + \textstyle\sum_{m \in \mathcal{B} \cup \mathcal{G} \setminus \{i\}} \zeta_{mi}^{max} \right), \\ &\forall i \in \mathcal{B} \cup \mathcal{G}, j \in \mathcal{A}_i, k \in \mathcal{L}_i \cap \mathcal{L}_j \end{aligned} \tag{3.5}$$

The right hand side is equal to 0 if MR i assigned channel k and the MC is assigned a channel. In this case, P_j^k must be set to a value that satisfies the required SINR threshold γ at MR i. On the other hand, if channel k is not assigned to MR i or MC j is not assigned a channel, then the inequality is satisfied for any positive (≥ 0) value of P_j^k using the same argument as before. Finally, the *RBA* problem can be formulated as an MILP as follows:

Maximize $\sum_{i \in \mathcal{A}} \sum_{k \in \mathcal{L}_i} c_i^k$, subject to:

(a) Channel assignment:

$$\sum_{k\in\mathcal{L}_j} c_j^k \leq 1,\ \forall\ j \in \mathcal{B}\cup\mathcal{G}\cup\mathcal{A}. \tag{3.6}$$

$$\sum_{k\in\mathcal{L}_j} c_j^k \leq \sum_{k\in\mathcal{L}_i} c_i^k,\ \forall\ i \in \mathcal{B}\cup\mathcal{G},\ j\in\mathcal{A}_i \tag{3.7}$$

(b) Upstream connectivity constraints:

$$\sum_{j:(i,j)\in E} f_{ij} - \sum_{j:(j,i)\in E} f_{ji} = 0,\ i\in\mathcal{B}\cup\mathcal{G}. \tag{3.8}$$

$$\sum_{j:(s,j)\in E} f_{sj} = \sum_{j:(j,d)\in E} f_{jd} \tag{3.9}$$

$$f_{sj} = \sum_{i\in\mathcal{A}_j}\sum_{k\in\mathcal{L}_i} c_i^k,\ j\in\mathcal{B}\cup\mathcal{G}. \tag{3.10}$$

$$f_{ij} \leq |\mathcal{A}|\cdot \sum_{k\in\mathcal{L}_i\cap\mathcal{L}_j} c_j^k,\quad (i,j)\in E. \tag{3.11}$$

(c) Downstream connectivity constraints:

$$\sum_{j:(i,j)\in\overline{E}} g_{ij} - \sum_{j:(j,i)\in\overline{E}} g_{ji} = 0,\ i\in\mathcal{B}\cup\mathcal{G}. \tag{3.12}$$

$$\sum_{j:(\overline{s},j)\in\overline{E}} g_{\overline{s}j} = \sum_{j:(j,\overline{d})\in\overline{E}} g_{j\overline{d}} \tag{3.13}$$

$$g_{j\overline{d}} = \sum_{i\in\mathcal{A}_j}\sum_{k\in\mathcal{L}_i} c_i^k,\ j\in\mathcal{B}\cup\mathcal{G}. \tag{3.14}$$

$$g_{ij} \leq |\mathcal{A}|\cdot \sum_{k\in\mathcal{L}_i\cap\mathcal{L}_j} c_j^k,\quad (i,j)\in\overline{E}. \tag{3.15}$$

(d) Power control constraints:

$$P_i^k \le P_r^{max} \cdot \sum_{j:\{j \in \mathcal{A}_i, k \in \mathcal{L}_j\}} c_j^k, \ \forall\ i \in \mathcal{B} \cup \mathcal{G}, k \in \mathcal{L}_i \tag{3.16}$$

$$P_i^k \le P_r^{max}, \ \forall\ i \in \mathcal{B} \cup \mathcal{G}, k \in \mathcal{L}_i \tag{3.17}$$

$$P_j^k \le P_c^{max} \cdot c_i^k, \quad \forall\ i \in \mathcal{B} \cup \mathcal{G}, j \in \mathcal{A}_i,\ k \in \mathcal{L}_i \cap \mathcal{L}_j \tag{3.18}$$

(e) Inter-cell interference:

$$\begin{aligned} \zeta_{ij}^k \ge P_i^k \Psi_{ij}^k, \quad & \forall i \in \mathcal{B} \cup \mathcal{G},\ j \in \mathcal{A} \cup \mathcal{B} \cup \mathcal{G} \backslash (\mathcal{A}_i \cup \{i\}), \\ & k \in \mathcal{L}_i \cap \mathcal{L}_j. \end{aligned} \tag{3.19}$$

$$\begin{aligned} \zeta_{ij}^k \ge P_m^k \Psi_{mj}^k, \quad & \forall i \in \mathcal{B} \cup \mathcal{G},\ j \in \mathcal{A} \cup \mathcal{B} \cup \mathcal{G} \backslash (\mathcal{A}_i \cup \{i\}), \\ & m \in \mathcal{A}_i,\ k \in \mathcal{L}_m \cap \mathcal{L}_j. \end{aligned} \tag{3.20}$$

(f) Link reliability constraints:

$$\begin{aligned} & \Psi_{ij}^k P_i^k - \gamma \left(N_0 + \textstyle\sum_{m \in \mathcal{B} \cup \mathcal{G} \backslash \{i\}} \zeta_{mj}^k \right) \ge \\ & \gamma (c_j^k - 1) \left(N_0 + \textstyle\sum_{m \in \mathcal{B} \cup \mathcal{G} \backslash \{i\}} \zeta_{mj}^{max} \right), \\ & \forall i \in \mathcal{B} \cup \mathcal{G}, j \in \mathcal{A}_i, k \in \mathcal{L}_i \cap \mathcal{L}_j \end{aligned} \tag{3.21}$$

$$\begin{aligned} & \Psi_{ji}^k P_j^k - \gamma \left(N_0 + \textstyle\sum_{m \in \mathcal{B} \cup \mathcal{G} \backslash \{i\}} \zeta_{mi}^k \right) \ge \\ & \gamma (c_i^k + \textstyle\sum_{w \in \mathcal{L}_j} c_j^w - 2) \left(N_0 + \textstyle\sum_{m \in \mathcal{B} \cup \mathcal{G} \backslash \{i\}} \zeta_{mi}^{max} \right), \\ & \forall i \in \mathcal{B} \cup \mathcal{G}, j \in \mathcal{A}_i, k \in \mathcal{L}_i \cap \mathcal{L}_j \end{aligned} \tag{3.22}$$

The above two equations are the same as equations (3.4) and (3.5), and are repeated here for clarity.

3.3.2 Transmitter-based channel allocation (TBA)

We refer to the channel allocation strategy that follows the FT-TR mode as the *transmitter-based channel allocation* (TBA). In TBA, a node (MC or MR) is assigned a fixed channel to transmit on, while it can receive on any of the channels available to it. As explained earlier, this strategy requires the existence of a CCC so that the transmitter node can make the receiver node tune to its channel. Therefore, we study this allocation strategy under two assumptions: first, a preassumed CCC exists; second, the existence of a CCC depends on channel availability and it is not preassumed. The *TBA problem* is defined similar to the *RBA problem* except that the $FT-TR$ is used instead of the $TT-FR$ approach. The ILP formulation of this case is presented in Appendix C.1.

3.3.3 All-tunable channel allocation (ATA)

Following the TT-TR mode, we propose the *All-tunable channel allocation* (ATA) strategy, under which channels are assigned to links rather than nodes. Therefore, an MR might have to listen/transmit on different channels. As for the MC, it will have to receive on one channel (the one assigned to the downlink from the MR to the MC), and transmit on one channel (the one assigned to the uplink from the MC to the MR). We also study this allocation strategy under two assumptions: first, a preassumed CCC exists; second, the existence of a CCC depends on channel availability and it is not preassumed. The *ATA problem* is defined similar to the *RBA problem* except that TT-TR mode is used. The MILP formulation of the ATA problem is presented in Appendix C.2. Table 3.1 summarizes the differences between the three channel allocation strategies.

3.4 Heuristic Solution for RBA

As our results, presented in Section 3.6, imply the superiority of RBA strategy over other allocation strategies, we propose in this section a heuristic solution for the RBA problem. We solve the problem in three phases:

Table 3.1: Differences between the RBA, TBA, and ATA channel allocation strategies

	RBA		TBA		ATA	
	Rx Ch.	Tx Ch.	Rx Ch.	Tx Ch.	Rx Ch.	Tx Ch.
MR	fixed	tunable	tunable	fixed	tunable	tunable
MC	fixed	fixed	fixed	fixed	fixed	fixed
CCC	No		Yes		Yes	

(1) *Channel assignment to MRs*: in this phase, MRs are assigned channels such that their upstream and downstream connectivity with the gateway(s) is maintained.

(2) *Finding the maximum number of reliable uplinks*: based on the channel assignment made in the first phase, and the channel availability at each MC, we assign transmission powers to MCs such that the number of reliable uplinks (MC→MR) is maximized. This power assignment is achieved using the power control algorithm we propose in Subsection 3.4.2.

(3) *Channel assignment to MCs*: to the MCs that have reliable uplinks after phase (2), we allocate channels and transmission powers such that the number of reliably served MCs is maximized. An MC is reliably served if both its uplink and downlink are served within the reliability threshold γ.

3.4.1 Phase 1: Channel assignment to MRs

The first phase in our solution to the RBA problem is to allocate channels to MRs such that the upstream and downstream connectivity with the backbone network is established for the maximum number of MRs. Before introducing the algorithm, we first give some definitions:

- $\mathbf{L}_{(\mathbf{t})}$ is an $N_r \times K$ matrix where N_r is the total number of MRs, and K is the total number of channels available in the system. This matrix represents channels that an MR can transmit on, hence the $(\mathbf{t})$ is the subscript of $\mathbf{L}$. Thus, the $(i,k)^{th}$ element of $\mathbf{L}_{(\mathbf{t})}$ is defined as,

$$\mathbf{L_{(t)}}[i,k] = \begin{cases} 1 & \text{if } k \in \mathcal{L}_i \\ 0 & \text{otherwise} \end{cases} \tag{3.23}$$

- $\mathbf{L_{(r)}}$ is an $N_r \times K$ matrix that represents the channels that an MR can receive on. Although this matrix is initially the same as $\mathbf{L_{(t)}}$, it will become different when MRs are assigned channels to receive on as we will see later. The $(i,k)^{th}$ element of $\mathbf{L_{(r)}}$ is initially defined as,

$$\mathbf{L_{(r)}}[i,k] = \begin{cases} 1 & \text{if } k \in \mathcal{L}_i \\ 0 & \text{otherwise} \end{cases} \tag{3.24}$$

- $\mathbf{I}$ is an $N_r \times N_r$ matrix that represents the accessibility between MRs. In other words,

$$\mathbf{I}[m,n] = \begin{cases} 1 & \text{if } \exists\ k \in \mathcal{L}_m \cap \mathcal{L}_n : \Psi_{mn}^{k} P_r^{max} \geq P_{th} \\ 0 & \text{otherwise} \end{cases} \tag{3.25}$$

- $\mathbf{W}$ is an $N_r \times K$ matrix that represents the weights of assigning channels to MRs. The element $\mathbf{W}[i,k]$ is the weight of assigning channel k to MR i defined as the number of MCs in $\mathcal{A}_i$ that can access i on channel k.

- $\mathbf{C}_{(u)}$ is a row-vector of length N_r such that $\mathbf{C}_{(u)}[i]$=1 if there exists a directed path from MR i to the gateway[3], and equals 0 otherwise. This connectivity is evaluated assuming that i is assigned all the channels that have their values in the row-vector $\mathbf{L}_{(r)}[i,*]$ set to 1.

- $\mathbf{C}_{(d)}$ is a row-vector of length N_r such that $\mathbf{C}_{(d)}[i]$=1 if there exists a directed path from the gateway to MR i, and equals 0 otherwise. Similar to $\mathbf{C}_{(u)}$, this connectivity is evaluated assuming that i is assigned all the channels with their values in the row-vector $\mathbf{L}_{(r)}[i,*]$ set to 1.

[3]We assume that a single gateway exists in the system. However, the proposed algorithm can be easily extended to the case of multiple gateways by assuming a hypothetical gateway that has all the channels available and is connected to all the actual gateways.

- Define:

$$\underset{x}{\operatorname{argmax}}\ (f(x), g(x)) := \{x \mid \forall y : f(y) \leq f(x), \text{ and if } f(y) = f(x) \text{ then } g(y) \leq g(x)\}$$

The *Routers Channel Allocation* (RCA) algorithm is outlined in Algorithm 1. First, the matrices $\mathbf{L}_{(t)}$ and $\mathbf{L}_{(r)}$ are calculated using equations (3.23) and (3.24) respectively. These two matrices are then used to evaluate the upstream and downstream connectivity vectors $\mathbf{C}_{(u)}$ and $\mathbf{C}_{(d)}$ respectively (line 2). The algorithm, then, operates iteratively and selects one MR at each iteration for processing. The gateway is selected first, then MRs are selected in breadth first manner based on their connectivity with already processed MRs. Let MR i be the one selected at the current iteration. If there exists a subset of channels $\mathcal{L}_i^* \subseteq \mathcal{L}_i$ such that each channel of which preserves (if assigned alone to MR i) the connectivity in $\mathbf{C}_{(u)}$ and $\mathbf{C}_{(d)}$ (lines 6-11), then MR i must be assigned a channel from $\mathcal{L}_i^*$. From $\mathcal{L}_i^*$, channels that were not assigned to adjacent cells are preferred over other channels. The channel $\hat{k}$ with the maximum weight, i.e., $\mathbf{W}[i, \hat{k}]$, is selected, and ties are broken based on the number of MRs that can access MR i on $\hat{k}$.

If subset $\mathcal{L}_i^*$ is empty (which means that there is no channel that if assigned to MR i, the connectivity will be preserved), then the channel that allows the maximum number of neighboring MRs to access MR i is selected (lines 12-13).

After allocating a channel to MR i, the connectivity vectors $\mathbf{C}_{(u)}$ and $\mathbf{C}_{(d)}$ are updated. Also, all the MCs that belong to $\mathcal{A}_i$ and cannot access MR i on the selected channel $\hat{k}$ must be removed. Moreover, all MCs that have their parent MRs disconnected from the backbone network (either in the upstream or the downstream direction) must be removed.

Algorithm 1: RCA: Routers Channel Allocation algorithm

input : $\mathcal{T} = \mathcal{B} \cup \mathcal{G}$, $\mathcal{L}_i \; \forall \; i \in \mathcal{B} \cup \mathcal{G}$
output : channel assignment matrix $\mathbf{L_{(r)}}$.

Calculate $\mathbf{L_{(t)}}$ and $\mathbf{L_{(r)}}$ using equations (3.23) and (3.24) respectively;
Evaluate upstream and downstream connectivity ($\mathbf{C}_{(u)}$ and $\mathbf{C}_{(d)}$ respectively);
repeat
  Pick up an MR i from $\mathcal{T}$;
  Let $\mathcal{L}_i^* \subseteq \mathcal{L}_i$ be the subset of channels from $\mathcal{L}_i$ such that if any of these channels is assigned to MR i, the connectivity in $\mathbf{C}_{(u)}$ and $\mathbf{C}_{(d)}$ will be preserved;
  if $\mathcal{L}_i^* \neq \emptyset$ **then**
    Let $\mathcal{S} \subseteq \mathcal{L}_i^*$ be the subset of channels from $\mathcal{L}_i^*$ that are not assigned to any MR of the cells adjacent to cell i;
    if $\mathcal{S} \neq \emptyset$ **then**
      $\hat{k} = \underset{k \in \mathcal{S}}{\text{argmax}} \, (\mathbf{W}[i,k], \sum_{j=0}^{N_r} \mathbf{I}[i,j] \cdot \mathbf{L_{(t)}}[j,k])$
    else
      $\hat{k} = \underset{k \in \mathcal{L}_i^*}{\text{argmax}} \, (\mathbf{W}[i,k], \sum_{j=0}^{N_r} \mathbf{I}[i,j] \cdot \mathbf{L_{(t)}}[j,k])$;
  else
    $\hat{k} = \underset{k \in \mathcal{L}_i}{\text{argmax}} \, (\mathbf{I}[i,*] \times \mathbf{L_{(t)}}[*,k])$;
  $\mathbf{L}_{(r)}[i,*] = \overrightarrow{1}_{\hat{k}}$;
  Update $\mathbf{C}_{(u)}$ and $\mathbf{C}_{(d)}$;
  forall $w \in \mathcal{B}$ **do**
    if $\mathbf{C}_{(u)}[w] = 0$ *or* $\mathbf{C}_{(d)}[w] = 0$ **then**
      $\mathcal{A}_w = \emptyset$;
  forall $j \in \mathcal{A}_i$ **do**
    if $\hat{k} \notin \mathcal{L}_j$ **then**
      $\mathcal{A}_i = \mathcal{A}_i \backslash \{j\}$;
  $\mathcal{T} = \mathcal{T} \backslash \{i\}$;
until $\mathcal{T} = \emptyset$;
return $\mathbf{L_{(r)}}$;

3.4.2 Phase 2: Finding the maximum number of reliable uplinks

Before we move into the second phase of our solution strategy, we propose a power control algorithm (PCA). The PCA algorithm takes as an input two sets of links: a set of uplinks $\mathcal{Q}_u(k)$ and a set of downlinks $\mathcal{Q}_d(k)$, as well as the channel k on which those links operate. If there exists a power allocation for all links' transmitters such that the SINR at all links' receivers is at least γ, then the algorithm returns 1, otherwise it returns 0. To test the existence of a feasible power allocation (one that achieves the reliability of all links), we propose a simple linear programming (LP) formulation that aims at finding any feasible solution, i.e., no optimization objective. The LP is outlined in Algorithm 2. The first and the second constraints correspond to the interference caused by active cells at the receivers in other active cells (similar to the inter-cell constraints (3.19) and (3.20) in Section 3.3. An active cell is a cell that has at least one link in $\mathcal{Q}_u(k) \cup \mathcal{Q}_d(k)$. The third and the fourth constraints, on the other hand, correspond to the reliability requirement of the uplinks and downlinks respectively. $r(e)$ and $t(e)$ denote the receiver node and the transmitter node of link e respectively.

Now, we can explain our solution for the second phase, i.e., maximizing the number of reliable uplinks. This phase is outlined in lines [6-13] in Algorithm 3. The output of the first phase is the allocation of exactly one channel $k \in \mathcal{L}_i$ for each MR $i \in \mathcal{B} \cup \mathcal{G}$. The idea is to go over the channels in $\mathcal{L}$ one by one. For each channel k, we find the set of potential uplinks on channel k, denoted as $\mathcal{Q}_u(k)$, as shown in line 7. If $\mathcal{Q}_u(k)$ is not empty, then for each uplink e, we find the *maximum channel gain*, λ_e, between $t(e)$ and all the receiving MRs in $\mathcal{Q}_u(k)$ except $r(e)$ as follows:

$$\lambda_e = \max_{i:\exists(j,i)\in\mathcal{Q}_u(k), i\neq r(e)} \Psi^k_{t(e),i} \tag{3.26}$$

Then, we process the uplinks in $\mathcal{Q}_u(k)$ in ascending order of their λ_e values. For each uplink, we use the PCA algorithm to find out whether it can be supported, i.e., reliably served, without affecting, i.e., breaking the reliability of, already reliably served uplinks. If so, the uplink is added to the set of reliable uplinks $\mathcal{Q}^r_u(k)$, otherwise it will not be added.

Algorithm 2: PCA: Power control Algorithm

input : $\mathcal{Q}_u(k)$, $\mathcal{Q}_d(k)$, channel k
output : An integer in $\{0, 1\}$.
// Find the set of active, being an uplink receiver or a downlink transmitter for at least one link in $\mathcal{Q}_u(k) \cup \mathcal{Q}_d(k)$*, MRs* $\bar{\mathcal{B}}$.

$\bar{\mathcal{B}} := \{i : \exists e = (i, j) \in \mathcal{Q}_d(k) \mid e = (j, i) \in \mathcal{Q}_u(k)\}$;

// For each MR i*, find the subset* $\bar{\mathcal{A}}_i \subset \mathcal{Q}_u(k) \cup \mathcal{Q}_d(k)$ *of links that do not belong to the cell managed by* i.

$\bar{\mathcal{A}}_i := \{e \in \mathcal{Q}_u(k) : r(e) \neq i\} \cup \{e \in \mathcal{Q}_d(k) : t(e) \neq i\}$;

Solve the following LP:

$Maximize\ 0$
Subject to:

$$\Psi^k_{t(e_1),r(e_2)} P^k_{t(e_1)} \leq \zeta^k_{r(e_1),r(e_2)},\ \forall e_1 \in \mathcal{Q}_u(k), e_2 \in \bar{\mathcal{A}}_{r(e_1)};$$
$$\Psi^k_{t(e_1),r(e_2)} P^k_{t(e_1)} \leq \zeta^k_{t(e_1),r(e_2)},\ \forall e_1 \in \mathcal{Q}_d(k), e_2 \in \bar{\mathcal{A}}_{t(e_1)};$$
$$\Psi^k_{t(e),r(e)} P^k_{t(e)} - \gamma N_0 - \gamma \sum_{m \in \bar{\mathcal{B}} \setminus \{r(e)\}} \zeta^k_{m,r(e)} \geq 0, \forall e \in \mathcal{Q}_u(k);$$
$$\Psi^k_{t(e),r(e)} P^k_{t(e)} - \gamma N_0 - \gamma \sum_{m \in \bar{\mathcal{B}} \setminus \{t(e)\}} \zeta^k_{m,r(e)} \geq 0, \forall e \in \mathcal{Q}_d(k);$$

if *the above LP has a feasible solution* **then**
  return 1;
else
  return 0;

3.4.3 Phase 3: Channel allocation to MCs

The last phase is channel allocation to MCs, i.e., downlinks. First of all, the MCs to be considered in the phase are only the ones that have reliable uplinks with their parent MRs after the second phase. Therefore, in lines [15-16] of Algorithm 3, we set $A_i\ \forall i \in \mathcal{B} \cup \mathcal{G}$ to those MCs that have reliable uplinks with MR i. Similar to what we did with uplinks, we need to process potential downlinks in ascending order of their maximum channel gains. However, the case now is different. Each MC may have several channels available, i.e., $\mathcal{L}_j > 1$ for $j \in \mathcal{A}$. This provides us with multiple choices for each downlink, in contrary to the uplink case where each uplink has only one choice, i.e., the channel assigned to the MR of that uplink. Therefore, for each MC j, we will find $|\mathcal{L}_j|$ maximum channel gains each on one of the channels in $\mathcal{L}_j$. Let $\mathcal{P}$ be the set of all possible $(MC,\ channel)$ pairs defined as follows:

$$\mathcal{P} = \{(i, k) : i \in \bigcup_{j \in \mathcal{B} \cup \mathcal{G}} \mathcal{A}_j,\ k \in \mathcal{L}_i\}. \tag{3.27}$$

Recall that this set is evaluated after removing MCs that cannot be served on the uplink. Therefore, all MCs represented by at least one pair in $\mathcal{P}$ have passed the second phase, i.e., can be served reliably on the uplink. Let $p(i)$ denote the parent MR of MC i. Then for each pair $(i,k) \in \mathcal{P}$, the maximum channel gain $\lambda_{(i,k)}$ is calculated as follows:

$$\lambda_{(i,k)} = \max\{\max_{j:(j,k)\in\mathcal{P},p(i)\neq p(j)} \Psi_{ij}^{k}, \max_{j\in\mathcal{B}\cup\mathcal{G}\setminus\{p(i)\}:\exists(m,j)\in\mathcal{Q}_u^r(k)} \Psi_{ij}^{k}\} \tag{3.28}$$

The above equation finds the *maximum channel gain* $\lambda_{(i,k)}$ on channel k between MC i and any other MC that has channel k available or a MR that was assigned channel k in the first phase. Then, we process the pairs in $\mathcal{P}$ in ascending order of their *maximum channel gains*. For each pair (i,k), we add the downlink $(p(i),i)$ to the current set of reliable downlinks on channel k, $\mathcal{Q}_d^r(k)$ (initially empty), and the uplink $(i,p(i))$ to the current set of reliable uplinks on channel k', $\mathcal{Q}_u^r(k')$ (which is initially empty) where k' is the channel assigned to $p(i)$, i.e, $\mathbf{L}_{(r)}[i,k'] = 1$. Using the PCA algorithm, if both the uplink and the downlink can be served reliably without breaking the reliability of any link in $\mathcal{Q}_d^r(k)$ and $\mathcal{Q}_u^r(k')$, then this MC is added to the set of reliable MCs $\mathcal{A}^r$ and the downlink and the uplink are admitted to the set $\mathcal{Q}_d^r(k)$ and $\mathcal{Q}_u^r(k')$ respectively. Otherwise, the two links will be removed from $\mathcal{Q}_d^r(k)$ and $\mathcal{Q}_u^r(k')$ and the MC will not be added to $\mathcal{A}^r$. Once an MC is added to $\mathcal{A}^r$ by one of its pairs, other pairs of this MC in $\mathcal{P}$ will be ignored. This process is presented in lines [21-34] in Algorithm 3. Finally, to find a final power allocation for MRs and MCs, we run the PCA algorithm once for each channel over the set of reliable uplinks and downlinks on that channel.

The algorithm that combines all the three phases together is presented in Algorithm 3, which we call the *heuristic receiver-based channel allocation* (HRBA) algorithm. In the next section, we outline some possible medium access solutions that can work with the proposed RBA strategy.

Algorithm 3: HRBA: Heuristic Receiver Based Channel Allocation Algorithm

input : $\mathcal{L}$; $\mathcal{B}$; $\mathcal{G}$; $\mathcal{A}_i \forall i \in \mathcal{B} \cup \mathcal{G}$; $\mathcal{L}_i \forall i \in \mathcal{B} \cup \mathcal{G} \cup \mathcal{A}$.
output : Set of reliable MCs $\mathcal{A}^r$; transmission powers; channel allocation to MRs $\mathbf{L}_{(r)}$; channels allocation to MCs $\bar{\mathbf{L}}_{(r)}$.

//Phase 1: allocate channels to MRs.
$\mathbf{L}_{(r)} = GRA(\mathcal{B} \cup \mathcal{G}, \mathcal{L}_i \forall i \in \mathcal{B} \cup \mathcal{G})$;
$\mathcal{R} = \emptyset$;
$P_i^k = 0, \ \forall i \in \mathcal{B} \cup \mathcal{G} \cup \mathcal{A}, \ k \in \mathcal{L}_i$;
//Phase 2: find the set of reliable uplinks.
forall $k \in \mathcal{L}$ **do**
  $\mathcal{Q}_u(k) = \{e : r(e) \in \mathcal{B} \cup \mathcal{G}, t(e) \in \mathcal{A}_{r(e)}, k \in \mathcal{L}_{t(e)}, \mathbf{L}_{(r)}[r(e), k] = 1\}$;
  if $\mathcal{Q}_u(k) \neq \emptyset$ **then**
    For $e \in \mathcal{Q}_u(k)$ find λ_e using equation (3.26);
    $\mathcal{Q}_u^r(k) = \emptyset$;
    forall $e \in \mathcal{Q}_u(k)$ *in ascending order of* λ_e **do**
      if $PCA(\mathcal{Q}_u^r(k) \cup \{e\}, \emptyset, k) = 1$ **then**
        $\mathcal{Q}_u^r(k) = \mathcal{Q}_u^r(k) \cup \{e\}$;

//Phase 3: allocate channels to MCs.
$\mathcal{A}_i = \emptyset, \ \forall i \in \mathcal{B} \cup \mathcal{G}$;
$\mathcal{A}_i = \{j : (j, i) \in \bigcup_{k \in \mathcal{L}} \mathcal{Q}_u^r(k)\}, \ \forall i \in \mathcal{B} \cup \mathcal{G}$;
Find the set $\mathcal{P}$ using equation (3.27);
For each pair (i, k) in $\mathcal{P}$ find $\lambda_{(i,k)}$ using equation (3.28);
$\mathcal{Q}_d^r(k) = \emptyset, \ \mathcal{Q}_u^r(k) = \emptyset$;
Let $\bar{\mathbf{L}}_{(r)}$ be an $|\mathcal{A}| \times K$ matrix initially set to 0;
forall $(i, k) \in \mathcal{P}$ *in ascending order of* $\lambda_{(i,k)}$ **do**
  if $i \in \mathcal{A}^r$ **then**
    continue;
  $k' := \{k : \mathbf{L}_{(r)}[p(i), k] = 1\}$;
  $\mathcal{Q}_d^r(k) = \mathcal{Q}_d^r(k) \cup \{(p(i), i)\}$;
  $\mathcal{Q}_u^r(k') = \mathcal{Q}_u^r(k') \cup \{(i, p(i))\}$;
  $x = PCA(\mathcal{Q}_u^r(k), \mathcal{Q}_d^r(k), k)$;
  $y = PCA(\mathcal{Q}_u^r(k'), \mathcal{Q}_d^r(k'), k')$;
  if x=1 ***and*** y=1 **then**
    $\mathcal{A}^r = \mathcal{A}^r \cup \{i\}$;
    $\bar{\mathbf{L}}_{(r)}[i, k] = 1$;
  else
    $\mathcal{Q}_d^r(k) = \mathcal{Q}_d^r(k) \backslash \{(p(i), i)\}$;
    $\mathcal{Q}_u^r(k') = \mathcal{Q}_u^r(k') \backslash \{(i, p(i))\}$;

//Find final power allocation
$P_i^k = 0, \ \forall i \in \mathcal{B} \cup \mathcal{G} \cup \mathcal{A}, \ k \in \mathcal{L}_i$;
forall $k \in \mathcal{L}$ **do**
  $PCA(\mathcal{Q}_u^r(k), \mathcal{Q}_d^r(k), k)$;
$\mathcal{A}_i = \mathcal{A}_i \cap \mathcal{A}^r, \ \forall i \in \mathcal{B} \cup \mathcal{G}$;
return $\mathcal{A}^r$; $P_i^k \ \forall i \in \mathcal{B} \cup \mathcal{G} \cup \mathcal{A}, \ k \in \mathcal{L}_i$; $\mathbf{L}_{(r)}$; $\bar{\mathbf{L}}_{(r)}$;

3.5 Medium Access Control (MAC)

There is a need to devise a medium access control mechanism that can work with the proposed receiver based channel allocation strategy. Designing such a mechanism is challenged by the three major issues which we have discussed earlier; namely, the broadcast deformation problem, the mutlichannel hidden node problem, and the deafness problem. In this section, we discuss some of the possible medium access solutions that can be adopted under the proposed RBA strategy.

3.5.1 Synchronized Hybrid MAC

A hybrid TDMA-CSMA/CA mechanism can be used to overcome all of the three aforementioned challenges. The basic idea is to assign each SU a time slot (the length of which should be designed to guarantee a certain success probability) during which it acts as a receiver only. At the beginning of the slot assigned to a particular SU, say i, all other SUs that want to communicate with i must tune to the channel assigned to i. Then, they contend to gain access to that channel using the traditional CSMA/CA with RTS/CTS handshake. Although such an approach overcomes all of the aforementioned challenges, it requires centralized scheduling of listening slots, which makes it inflexible especially with dynamically changing traffic rates.

3.5.2 Synchronized MAC Protocol For Multi-Hop Cognitive Radio Networks (Sync-MAC)

T.R. Kondareddy and P. Agrawal proposed a synchronized MAC protocol for multihop cognitive radio networks that does not require a common control channel [73]. This protocol assumes that nodes are equipped with two radio transceivers. One of the two radios is dedicated to listening to control signals, while the other is used for transmitting and receiving data. Each of the exploited channels is assigned dedicated time slot in a periodic fashion. The slot is long enough to exchange control signals to coordinate the communication between a pair of nodes (like RTS-CTS). When node A wants to transmit to node B, it chooses the slot of a common channel between the two, starts a backoff counter, and when the backoff finishes it sends an

RTS if the channel is idle. B replies with the CTS, and then the communication starts. Nodes that have heard either the RTS or the CTS will realize that both nodes will not be available for the NAV value (Network Allocation Vector). Note that no node will miss any control packets transmitted over a particular channel at any time, even if it is currently transmitting on another channel, because it has a dedicated radio for that purpose (i.e., listening to control messages). This protocol can be used for medium access control under the RBA allocation strategy. The only difference here is that each node can receive on one channel only (the one assigned to it under the RBA strategy). According to [73], this MAC protocol outperforms CCC based MAC in terms of both throughput and network connectivity.

The problem in using this MAC protocol for the specific network studied in this chapter is that it does not guarantee that multihop transmissions (between MRs) will not cause interference to single-hop transmissions (those between an MR and the MCs it serves). Our MILP formulations assumed disjointness between multihop and single hop transmissions. Another drawback is the two radios per node requirement. Therefore, we propose a modified version of the the Sync-MAC to tackle the specifics on the network studied in this chapter.

3.5.3 Synchronized MAC for RBA

We now present a modified version of the synchronous MAC protocol discussed in the previous subsection which does not require two radio transceivers at the MCs and at the same time protects upstream/downstream links within each cell from interference caused by multihop communication. Initially, MRs use the synchronized MAC in its original form proposed in [73], with one difference which is that the time is divided intro frames as shown in Figure 3.2. The purpose of dividing the time into equal frames is to control the interference between multihop and single hop links. An MR operates in two modes, the inter-cell (multihop) mode, and the intra-cell (single hop mode). In the former, it exchanges data with its neighboring MRs using the Sync-MAC discussed earlier. When an MR wants to exchange data with MCs, it switches into the intra-cell mode. This switching can be done only at the beginning of a frame to keep all intra-cell transmissions across cells aligned, and so easy to protect. Before switching into the intra-cell mode, an MR must inform the neighboring MRs about its intention to schedule a periodic intra-cell communication sub-frame to disseminate/collect data to/from MCs. The bandwidth allocated to intra-cell communication (i.e., the percentage of time allocated to intra-cell communication) depends on the number of MCs associated with that cell and can be regulated at the network level. The MR can later change the length or frequency of this intra-cell communication period depending on the activity of the MCs in the intended cell. Adjacent MRs will avoid doing inter-cell communication on a channel, say k, during the intra-cell communication period of an adjacent cell if the latter has an uplink/downlink that used channel k.

We now describe the intra-cell communication and how it works. To initiate the intra-cell period for the first time, the MR notifies the MCs individually on their channels with an INITIATE control packet. This packet synchronizes the MCs with their parent MR, informs them about when the period will start and about its length, and finally the order of their assigned control mini slots shown in Figure 3.2. Each MC is assigned one mini slot during which it informs the MR about the number of packets it has ready for transmission, including control packets. The MR then calculates a schedule for the current period, and polls the MCs individually. The polling starts with a poll packet transmitted by the MR on the channel assigned to the MC. The MC then replies with its data packet to the MR. Finally the MR sends back the ACK, and then polls the next MC. This operation is summarized in Figure 3.3.

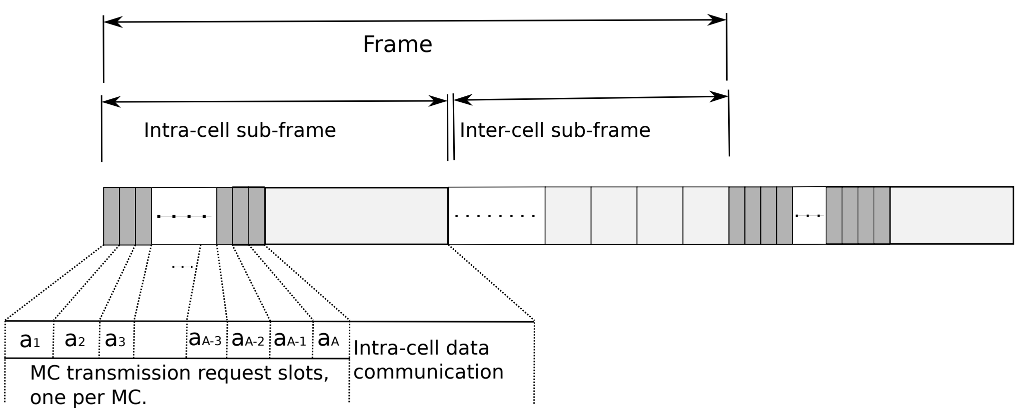


Figure 3.2: The format of the modefied Sync-MAC frame

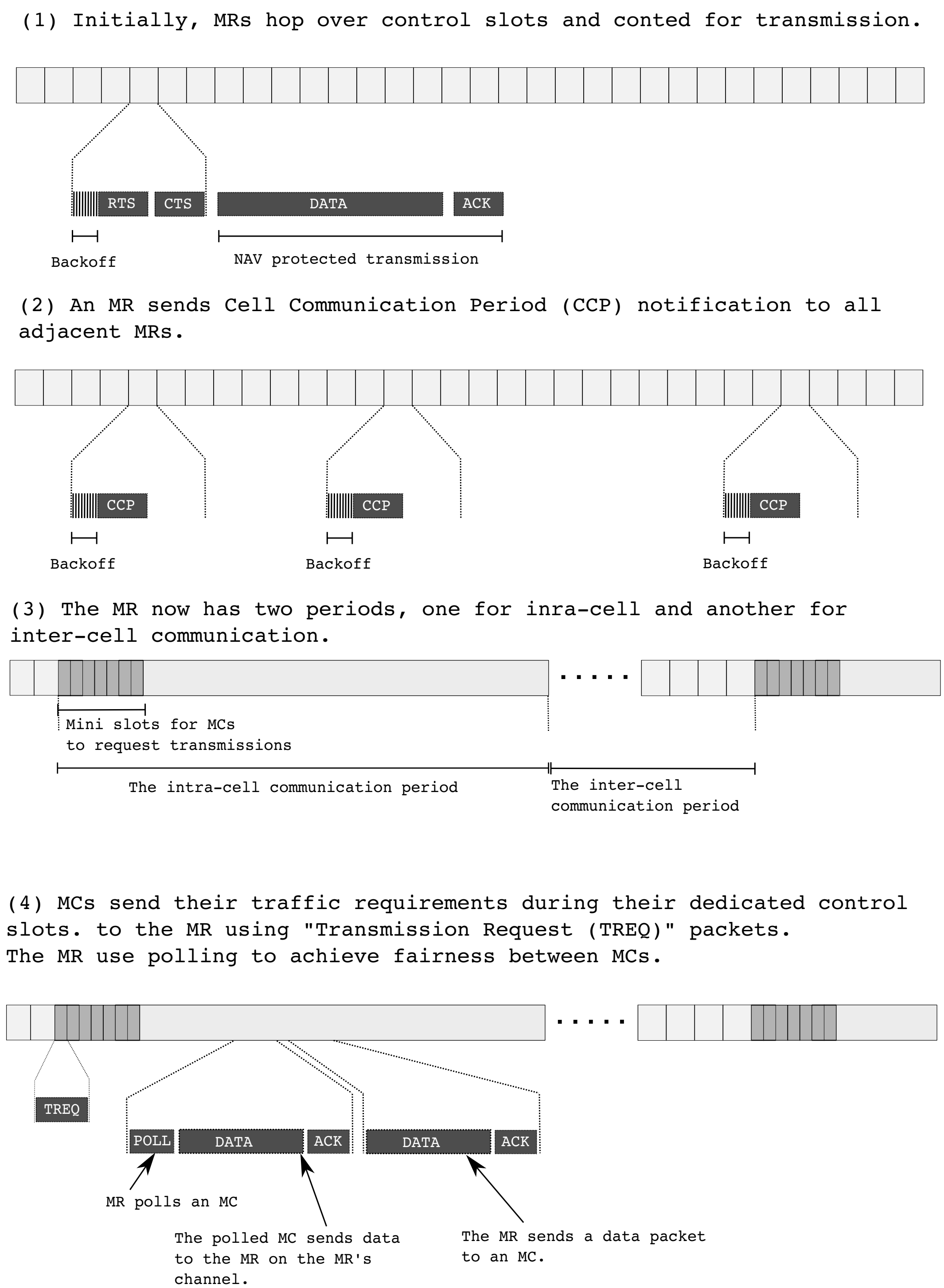


Figure 3.3: The operation of the MAC-RBA mechanism.

The performance of this MAC protocol depends on how the parameters (including the mini-slot length, the intra-cell sub-frame length, the inter-cell sub-frame length, etc.) are selected. To give an example that illustrates this dependency, we simulated a network of one MR and ten MCs with the parameters summarized in Table 3.2. We simulated unidirectional traffic (from MCs to MR) under different arrival rates and measured the aggregate throughput as well as the average latency (waiting time at each MC). The frame length is always fixed, while we vary the length of the intra-cell sub-frame. We also vary the packet mean arrival rate, and consider both cases of fixed and exponentially distributed packet inter-arrival time. The aggregate throughput under the fixed and exponentially distributed packet inter-arrival time are shows in figures 3.4 and 3.6 respectively, while the average delay results are shown in Figures 3.5 and 3.7 respectively. As the figure shows, the aggregate cell throughput depends on the amount of time allocated to that cell for intra-cell communication. Increasing the percentage of the frame time dedicated to intra-cell communication from 34.67% to 52% resulted in an increase in saturation throughput by $\approx 50\%$, and $\approx 200\%$ decrease in average delay.

Parameter	**Value**
Number of MCs	10
Frame length	27.55ms
Data packet size	512B
Data+Poll+ACK time	0.9 ms
Mini slot length	50us

Table 3.2: The simulation parameters of the MAC experiment

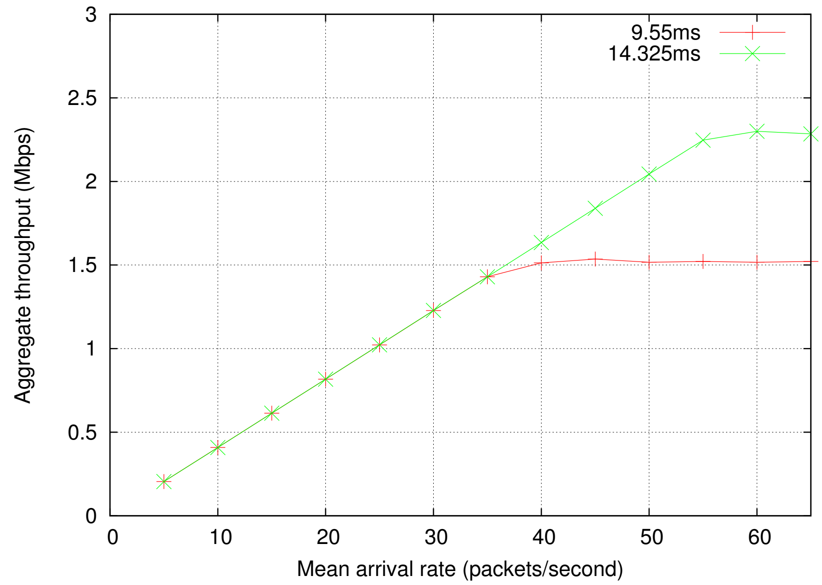


Figure 3.4: The aggregate throughput under different lengths of the intra-cell communication period, and fixed packet inter-arrival time.

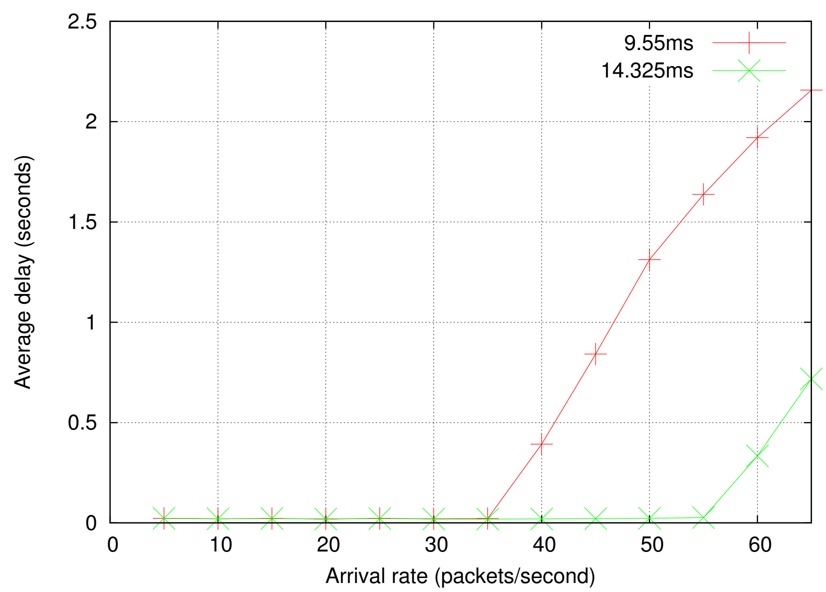


Figure 3.5: The average delay under different lengths of the intra-cell communication period, and fixed packet inter-arrival time.

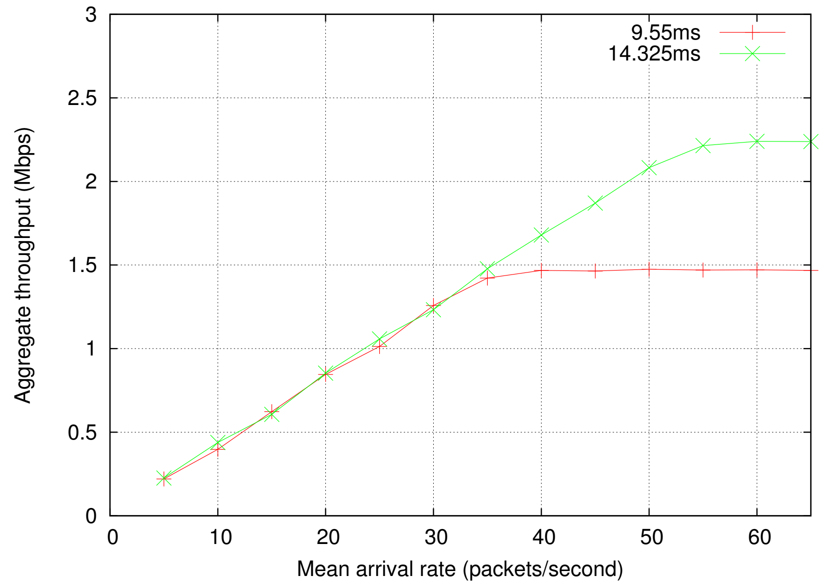


Figure 3.6: The aggregate throughput under different lengths of the intra-cell communication period, and exponentially distributed packet inter-arrival time.

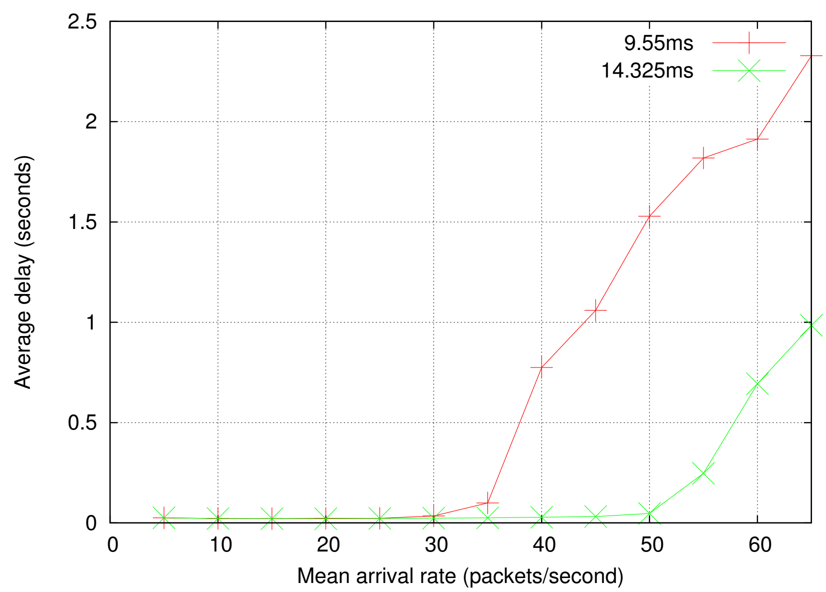


Figure 3.7: The average delay under different lengths of the intra-cell communication period, and exponentially distributed packet inter-arrival time.

3.6 Performance evaluation

In this section, we compare the optimal performance of all the three allocation strategies proposed in Section 3.3 in terms of the number of MCs served, and evaluate the performance of the proposed HRBA algorithm. To vary the channel availability distribution at different SUs, we vary the number and locations of PUs. The network is deployed in an $A \times A$ square area. The area is divided into N_r cells, such that $N_r = |\mathcal{B}| + |\mathcal{G}|$. We obtained the optimal solutions for $N_r = 4$, and 9 MRs and 100 MCs. In all scenarios, we assume the existence of a single gateway MR located at the right-bottom cell. The number of PUs is varied to achieve different channel availability distributions. Each PU is randomly assigned one of the K orthogonal channels available in the system. An SU (an MC or an MR) cannot use channel k if the user is less than R_p apart from an active PU that is assigned channel k. R_p is set equal to the cell radius, i.e., $R_p = \frac{A}{2\sqrt{N_r}}$. The maximum transmission power of an MR is calculated as $\left(\frac{2.5A}{2\sqrt{N_r}}\right)^{\alpha} N_0\gamma$ and of an MC as $\left(\frac{A}{\sqrt{N_r}}\right)^{\alpha} N_0\gamma$, where $\gamma = 15dB$ and $N_0 = 10^{-11}$ Watt. These values will guarantee that each MR will be able to reach the four adjacent (up, down, left, and right) MRs, and that each MC will be able to reach its parent MR. For all the experiments in this section, the path-loss exponent $\alpha = 3.76$.

3.6.1 Performance without a preassumed CCC

We first study the optimal performance of the three allocation strategies without presuming the existence of a CCC in the network. Figures 3.8 and 3.10 show the optimal performance of the three strategies for the case of 4 MRs and 9 MRs respectively. The number of active PUs is varied from 15 to 40 for the case of 4 MRs, and from 30 to 55 for the case of 9 MRs. Each point on each of the curves is the average of 100 randomly generated topologies. As the figures imply, the RBA approach outperforms the other two approaches. Notice that the difference in performance between RBA and ATA is higher for fewer PUs (i.e., the fewer the PUs the higher the channel availability). For instance, the number of served MCs using the RBA approach, in Figure 3.10, is on average 1.5 times that using the ATA approach for 30 PUs, however, this number jumps to 3.5 for the case of 55 PUs. The TBA approach, on the other hand, is always outperformed by the ATA approach, which is expected because they both require a CCC, but the ATA approach can use all the channels for transmission while the TBA approach is restricted to one channel only.

3.6.2 Performance with a preassumed CCC

In this subsection, we evaluate the performance of the three allocation strategies with the presumption that a CCC exists. We add one more channel and make it available to all nodes, i.e., no PU can use this extra channel. Figures 3.9 and 3.11 show the optimal performance of the three strategies for the case of 4 MRs and 9 MRs respectively. The number of PUs is varied from 15 to 40 for the case of 4 MRs, and from 30 to 55 for the case of 9 MRs. Each point in the two figures is the average performance of 100 randomly generated topologies. As the figures indicate, the RBA still outperforms the other two strategies even though a CCC is preassumed. However, the difference between RBA and the other approaches is less in this case than the case when no CCC is preassumed. The figures also show that for fewer PUs (which means high channel availability), the performance of the RBA strategy is very close to that of the ATA strategy.

3.6.3 Performance of the HRBA algorithm

In this subsection, we compare the performance of the HRBA algorithm to the optimal performance obtained using the MILP formulation in Section 3.3. In Figure 3.12, we show the number of served MCs (the average over a new, other than the experiments of Subsections 3.6.1 and 3.6.1, 100 randomly generated topologies) obtained using the HRBA algorithm and the MILP formulation for $|\mathcal{B}| = 8$, $|\mathcal{G}| = 1$, $|\mathcal{A}| = 100$ and $K = 6$. As the figure shows, the performance of the HRBA algorithm is close to the optimal solution, within with $\approx 14.7\%$ of the optimal (on average). Figure 3.13 shows the same results for $|\mathcal{B}| = 15$, $|\mathcal{G}| = 1$, $|\mathcal{A}| = 100$ and $K = 6$. Again, the HRBA algorithm is, on average, within $\approx 12\%$ of the optimal solution.

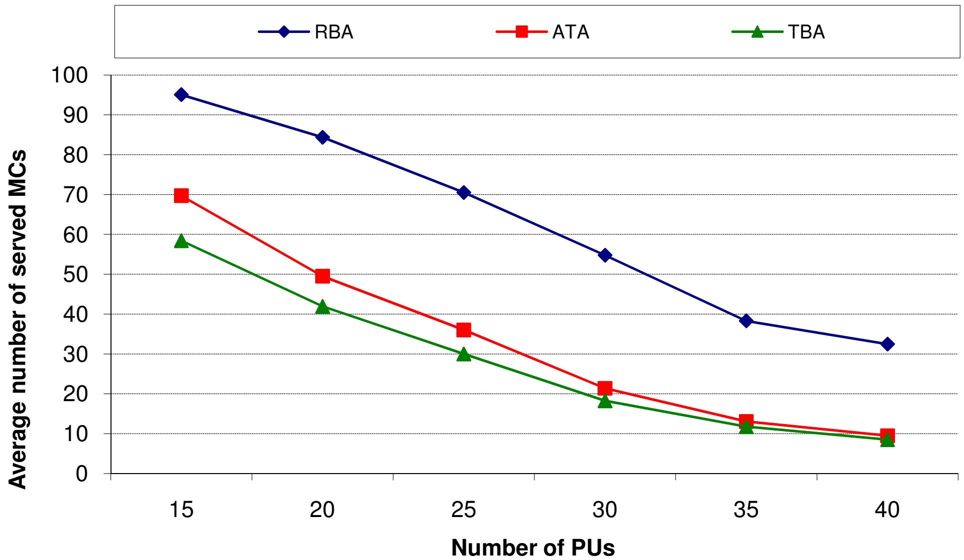


Figure 3.8: The performance of the three allocation strategies without a preassumed CCC. $|\mathcal{B}| = 3$, $|\mathcal{G}| = 1$, $|\mathcal{A}| = 100$, and $K = 6$.

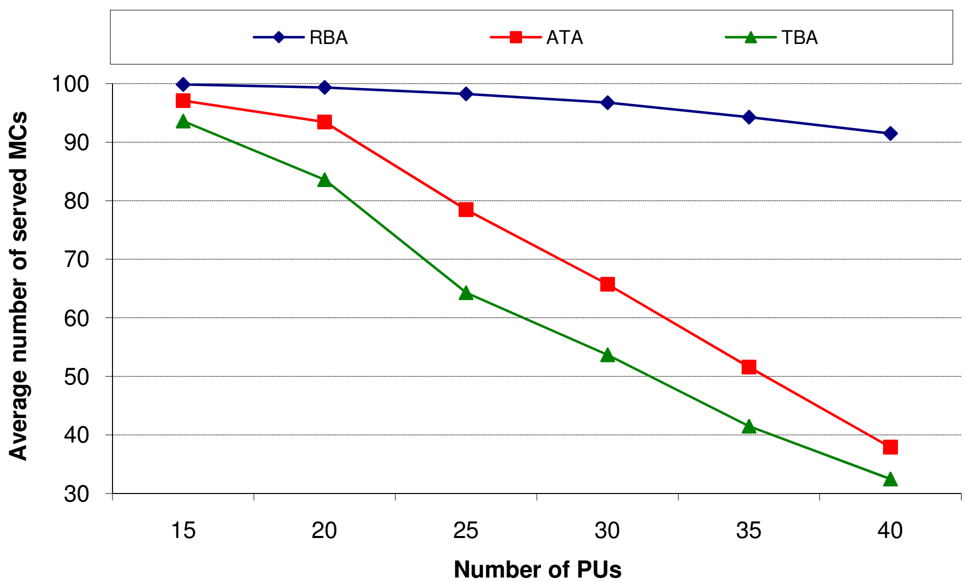


Figure 3.9: The performance of the three allocation strategies with a preassumed CCC. $|\mathcal{B}| = 3$, $|\mathcal{G}| = 1$, $|\mathcal{A}| = 100$, and $K = 7$.

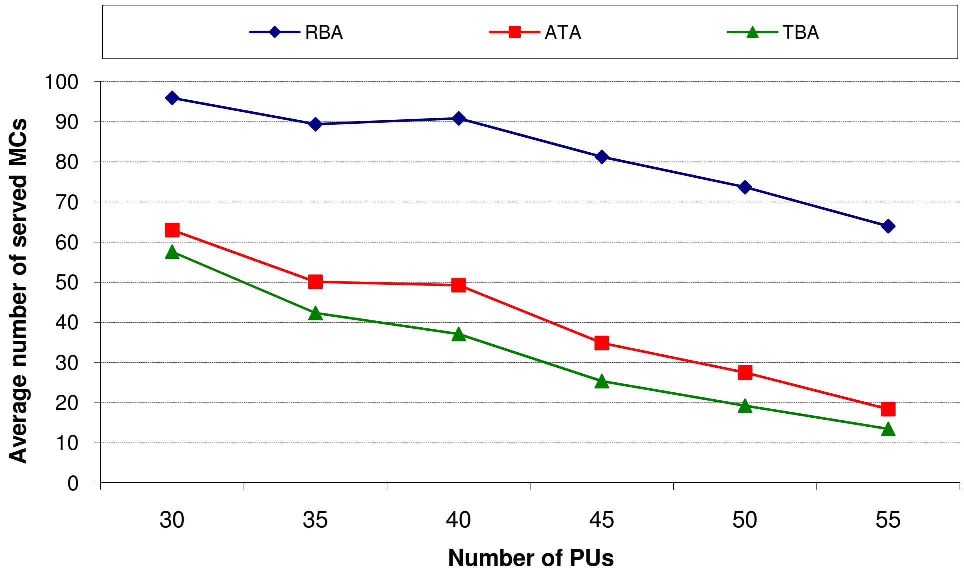


Figure 3.10: The performance of the three allocation strategies without a preassumed CCC. $|\mathcal{B}| = 8$, $|\mathcal{G}| = 1$, $|\mathcal{A}| = 100$, and $K = 6$.

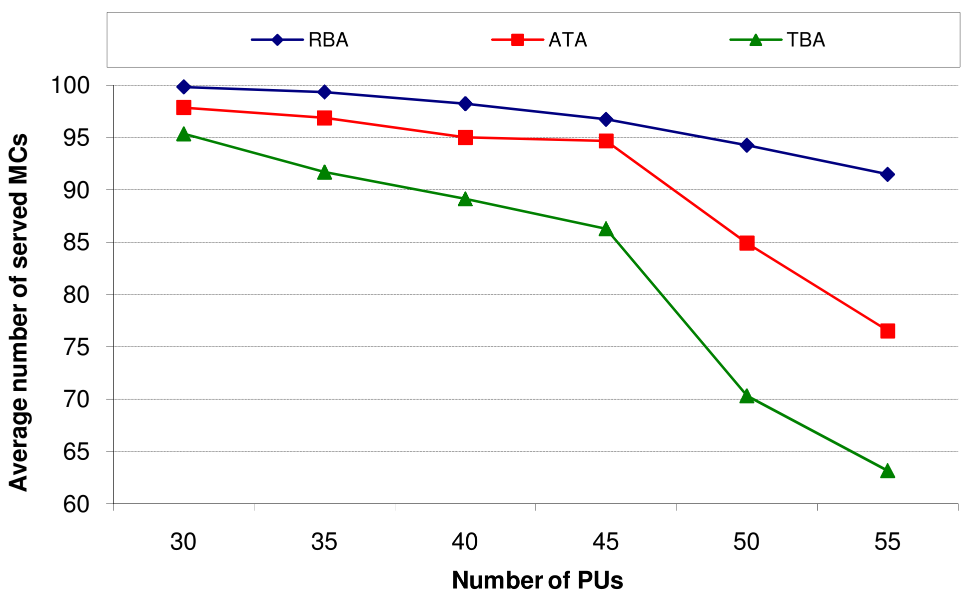


Figure 3.11: The performance of the three allocation strategies with a preassumed CCC. $|\mathcal{B}| = 8$, $|\mathcal{G}| = 1$, $|\mathcal{A}| = 100$, and $K = 7$.

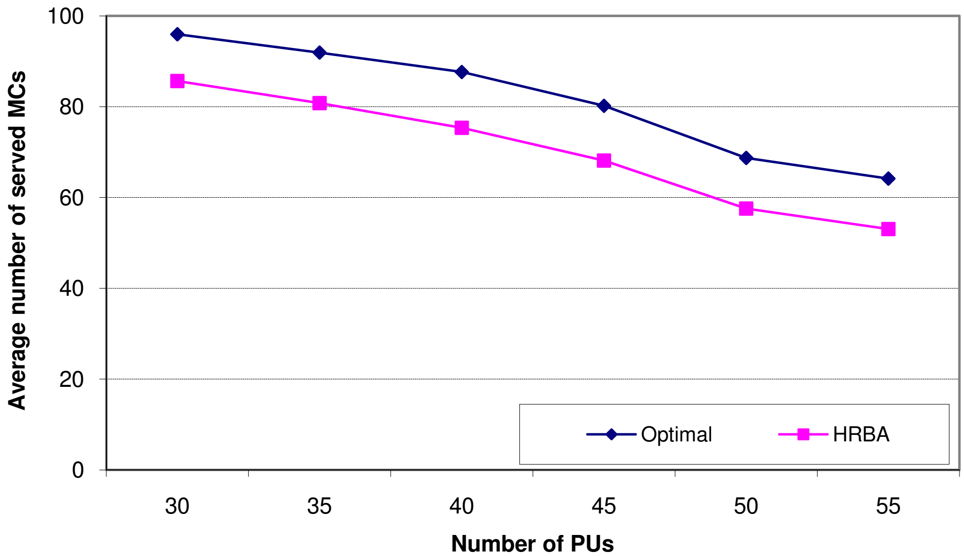


Figure 3.12: The performance of the HRBA algorithm compared to the optimal solution. $|\mathcal{B}| = 8$, $|\mathcal{G}| = 1$, $|\mathcal{A}| = 100$, and $K = 6$.

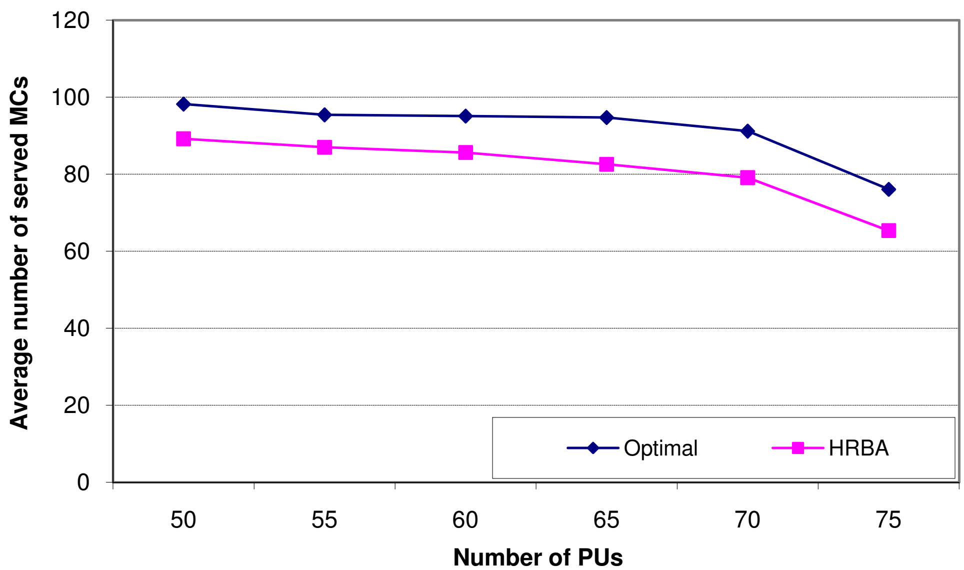


Figure 3.13: The performance of the HRBA algorithm compared to the optimal solution. $|\mathcal{B}| = 15$, $|\mathcal{G}| = 1$, $|\mathcal{A}| = 100$, and $K = 6$.

3.7 Chapter Summary

In this chapter, we studied the channel allocation problem in CR-WMNs. By controlling the tunability of the transmission and reception parts of the cognitive radio, four different modes of operation were defined for cognitive transceivers. Three channel allocation strategies based on the aforementioned modes were defined, namely receiver-based allocation *RBA*, transmitter-based allocation *TBA*, and all-tunable allocation *ATA*. MILP formulations were proposed RBA, TBA, and ATA strategies with the objective of maximizing the number of served MCs with a reliability guarantees on the uplink and downlink for each MC. Results show that the proposed RBA strategy outperforms the TBA and the ATA strategies even when a CCC is preassumed. We have also discussed a number of alternative MAC options which can be used with the proposed RBA strategy.

We also proposed a heuristic solution for the RBA problem. Results show that the accuracy of the proposed algorithm is, on average, within 28% of the optimal solution. We also outlined the possible MAC solutions that can work with the proposed RBA policy.

CHAPTER 4. EXPLOITING MULTICHANNEL DIVERSITY FOR COOPERATIVE MULTICAST

After focusing on the fundamental operational issues, like channel sensing, allocation, and management, over the past few years, recent studies in cognitive radio research have turned more to the application side, like security and video multicasting. In this chapter, we consider the multicast problem in cognitive radio wireless mesh networks (CR-WMNs). Due to the potential heterogeneity in channel availability among the members of a multicast group(s), the total multicast time may increase due to transmitting the multicast data over multiple channels. We propose, in this chapter, an assisted multicast strategy with the objective of minimizing the total multicast time. This assistance is composed of two main activities, first, allowing the receivers in a multicast group to forward the data they have received to other members of the multicast group(s), and second, allowing the transmission of coded (bitwise XORed) packets so that receivers belonging to different multicast groups can decode and extract their data concurrently. We show that the proposed assistance paradigm achieves a considerable reduction in the total multicast time, which in turn increases the system throughput. We formulate the problem of finding the optimal schedule for a single multicast group as an ILP (integer linear program), and then propose a heuristic algorithm that works for any number of multicast groups. The performance of the proposed heuristic is then evaluated against the optimal multicast schedule with no assistance from receiver nodes and no use of network coding.

We also study the issue of resolving potential conflict between the schedules of adjacent cells. Two solutions are proposed to guarantee collision-free schedule for the entire network; reactive collision resolution and proactive collision avoidance. Lastly, we propose a recovery algorithm to cope with the transmission failures due to PU activities. The performance of these solutions is also validated through simulation.

4.1 Introduction

In this chapter, we are interested in the multicast problem in cognitive radio wireless mesh networks (CR-WMNs) [14]. Generally speaking, a wireless mesh network consists of a number of mesh routers (MRs) each of which manages a group of mesh clients (MCs) forming a cell. MRs are connected through a gateway to a backbone network, like the Internet. An MC may reach the backbone network through its parent MR first, and then through multiple hops of MRs until reaching the gateway. The property of heterogeneous channel availability in cognitive radio networks may force the multicast process to take place over multiple channels, depending on the channel availability at different nodes. This may increase the time needed for a source to deliver the multicast data to all of its neighboring receivers. We study, in this chapter, the problem of minimizing the total multicast time in CR-WMNs by scheduling the multicast activity over both time and frequency and by also using the technique of network coding [53]. For this purpose, we propose a multicast mechanism that relies on three types of cooperation, namely; intra-group assistance, inter-group assistance, and the use of network coding. Three major operations facilitate these types of cooperation. The first operation is called *assistance*, in which some of the receiving members of a multicast group assist the multicast transmitter by forwarding the data on its behalf to other members of their group (or other groups). The second operation is called *overhearing*, in which some receiving members of a multicast group overhear the data destined to another group. This operation has two advantages; it first enables the inter-group assistance (forwarding) between different multicast groups, and also facilitates the delivery of multiple packets to different groups at the same time by using the third operation; the *codeword exchange operation*. In the *codeword exchange operation*, coded packets (bitwise XORed packets), which we refer to as codewords, are used in the assistance operation so that members of different multicast groups can decode and extract their own data using packets they have overheard from previous transmissions. Formal definitions of these operations will be given in Section 4.3.2.

4.1.1 The Multicast Scheduling Problem

Tremendous research has been conducted on multicast in multi-channel wireless networks to come up with efficient routing and/or channel allocation algorithms that maximize a number of different objectives. Energy-efficiency [74], spectrum efficiency [75, 76], throughput maximization [77, 78, 79, 80], and delay minimization [77, 81] are examples of these objectives. Multicasting in CRNs is different than that in traditional multi-channel wireless networks. In traditional multi-channel wireless networks, the same set of frequency channels is available at all nodes. This assumption may not hold in CRNs due to the *heterogeneity property* mentioned earlier, as illustrated in the example in Figure 4.1. The example shows three primary users p_1, p_2, and p_3, where each PU p_i utilizes frequency channel i. The gray grid-line circle around each PU represents the protection range of that PU, within which no SU is allowed to concurrently utilize (transmit or receive) the frequency channel with the PU. This range may be determined based on different criteria. One criterion, for example, is to guarantee certain bit-error-rate performance for PUs. Also, six SUs exist in the network where one of them acts as a multicast source, while the others act as multicast receivers. Note how the geographical distribution of the nodes affects the channel availability at different SUs (summarized by the set of channels shown besides each SU). This difference (or *heterogeneity*) forces the source SU to transmit the multicast data over the three frequency channels in order to cover all the multicast receivers. This summarizes the spatial part of the *heterogeneity property*.

The temporal part of the *heterogeneity property* is attributed to the channel usage distribution of PUs. For example, assume that at some point in time, p_2 is not using its frequency channel (i.e., channel 2). Then, all SUs will be able to use that channel, and consequently the source SU will be able to transmit the multicast data to all receivers over the same channel, i.e., channel 2. On the other hand, when the PU is back on the channel, SUs need to vacate it. This makes it a must for any scheduling algorithm to have a failure recovery plan because the return of the PU might break a some links in the previously calculated schedule. In addition

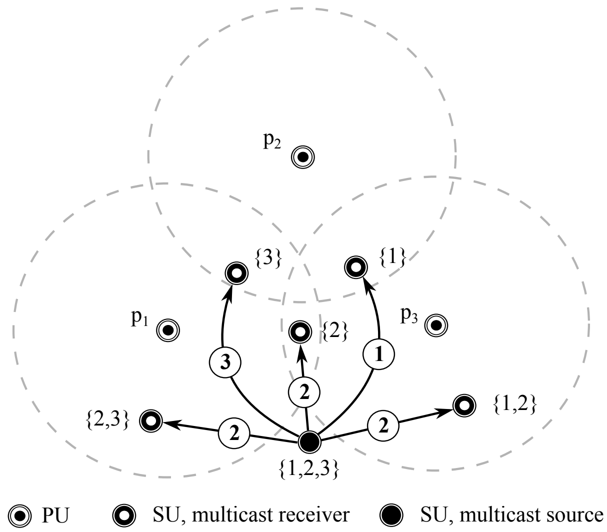


Figure 4.1: An example that illustrates the heterogeneity property of CRNs.

to its effect on the multicast throughput, in which we are interested in this chapter, the heterogeneity property leads to several other differences between CRNs and traditional wireless networks. We refer the reader to [82] for more information about this issue.

4.1.2 Contributions

We address the multicast scheduling problem in CR-WMNs in three stages. In the first phase, we study the scheduling of the multicast activity within a single cell. The contributions related to this phase are:

- *A scheduling strategy that exploits diversity in channel availability to enhance multicast throughput.*

- *A centralized implementation of the proposed scheduling strategy within a single cell.*

In the second phase, we study the issue of resolving potential conflict between the schedules of adjacent cells. We propose two solutions to guarantee collision-free schedule for the entire network; reactive collision resolution and proactive collision avoidance. In the last phase, we propose a recovery algorithm to cope with the transmission failures due to PU activities. We finally evaluate the performance of the proposed algorithms and the effect of different network parameters on the achievable gain.

4.1.3 Organization

The system model is presented in Section 4.2. In Section 4.3, we formally define the *assisted multicast scheduling* problem and present some motivational examples. Then, in Section 4.4, we elaborate on the problem complexity and propose ILP formulations for unassisted multicast scheduling and assisted multicast scheduling problems. A heuristic approach to solve the assisted multicast scheduling problem is proposed in Section 4.5. Resolving collisions between adjacent cells and recovering failed transmissions are studied in sections 4.6 and 4.7 respectively. We evaluate the performance of the proposed algorithms in Section 4.8, and conclude in Section 4.9.

4.2 System Model

We consider a time-slotted CR-WMN that consists of a number of mesh routers (MRs) connected in multiple hops to a gateway MR(s) that provides access to the backbone network. The network is synchronized and operates in frames of time slots. Each MR, including the gateway MR, manages a set of mesh client (MCs) forming a cell. For cell i, let $\mathcal{A}_i = \{a_{0,i}, a_{1,i}, \cdots, a_{A_i,i}\}$ be the set of nodes (the MR and the MCs) in that cell, where A_i is the total number of MCs in the cell (i.e., $A_i = |\mathcal{A}_i| - 1$ as $a_{0,i}$ is MR and the rest are MCs). The only way for an MC to access the backbone network is through its parent MR. The CR-WMN coexists with a primary network that utilizes a set of orthogonal channels $\mathcal{L}$.

Secondary users (MRs and MCs) obtain the set of available channels (those which can be used without harming the primary network) through spectrum sensing using any of the techniques proposed in literature [83, 40, 84]. In this chapter, we assume that the channel availability at secondary users is quasi-static, i.e., does not change in a short period of time. This assumption is more suited to situations where PUs do not change their operating frequency very often. We further assume the presence of a common control channel on which nodes can exchange control information. Let $\mathcal{L}_i \subseteq \mathcal{L}$ denote the set of available data channels at node (MR or MC) i, where $\mathcal{L}$ is the set of all channels in the system. Furthermore, each MC must share at least one data channel with its parent MR to be serviced by the MR. Lastly, we assume that each node is equipped with only one cognitive radio for data transmission.

We treat the multicast process as a two-stages process. The first stage is to deliver the multicast data from the gateway to the MRs which have some of their MCs subscribing to the intended multicast session(s). The second stage is for each MR to deliver the received multicast data to the subscribing MCs within its cell. In this chapter, we are concerned with the second stage.

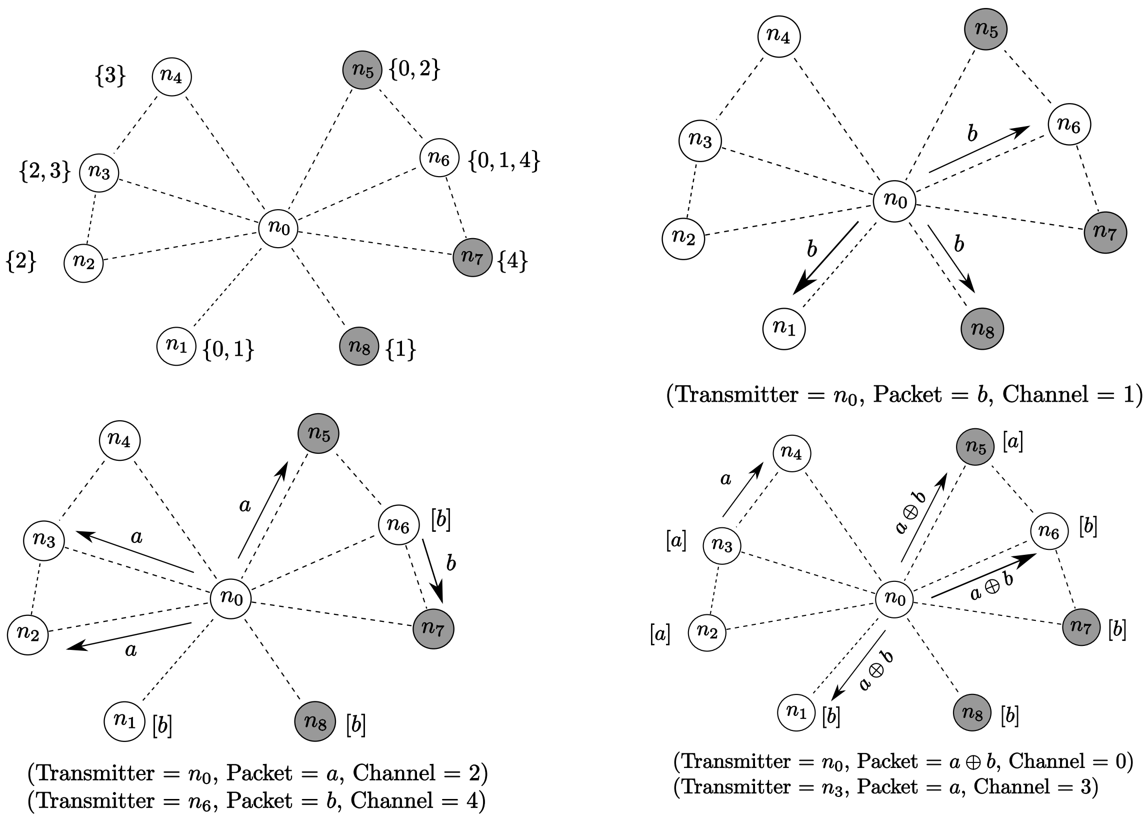


Figure 4.2: An example that shows the benefit of using assisted multicast in reducing the total multicast period.

Table 4.1: Enhancing throughput by introducing different assistance mechanisms

Scenario	Tx/Rx	T_1	T_2	T_3	T_4	T_5	T_6
Without assis.	Tx	$(n_0, a, 0)$	$(n_0, a, 2)$	$(n_0, a, 3)$	$(n_0, b, 0)$	$(n_0, b, 4)$	$(n_0, b, 1)$
	Rx	$(\{n_1, n_6\}, a, 0)$	$(\{n_2, n_3\}, a, 2)$	$(n_4, a, 3)$	$(n_5, b, 0)$	$(n_7, b, 4)$	$(n_8, b, 1)$
Intra-group assis.	Tx	$(n_0, a, 0)$	$(n_0, a, 2)$	$(n_3, a, 3)$, $(n_0, b, 0)$	$(n_0, b, 4)$	$(n_0, b, 1)$	-
	Rx	$(\{n_1, n_6\}, a, 0)$	$(\{n_2, n_3\}, a, 2)$	$(n_4, a, 3)$, $(n_5, b, 0)$	$(n_7, b, 4)$	$(n_8, b, 1)$	-
Inter-group assis.	Tx	$(n_0, b, 1)$	$(n_0, a, 2)$, $(n_6, b, 4)$	$(n_3, a, 3)$, $(n_0, b, 0)$	$(n_0, a, 0)$	-	-
	Rx	$(\{n_8, n_6\}, b, 1)$	$(\{n_2, n_3\}, a, 2)$, $(n_7, b, 4)$	$(n_4, a, 3)$, $(n_5, b, 0)$	$(\{n_1, n_6\}, a, 0)$	-	-
Codeword exchange	Tx	$(n_0, b, 1)$	$(n_0, a, 2)$, $(n_6, b, 4)$	$(n_3, a, 3)$, $(n_0, a \oplus b, 0)$	-	-	-
	Rx	$(\{n_1, n_6, n_8\}, b, 1)$	$(\{n_2, n_3, n_5\}, a, 2)$ $(n_7, b, 4)$	$(n_4, a, 3)$, $(\{n_1, n_5, n_6\}, a \oplus b, 0)$	-	-	-

4.3 Motivation and Problem Definition

Before we formally define the assisted multicast problem, we would like to present an example that illustrates the motivation behind this work, and then give some definitions.

4.3.1 Motivational Example

Consider the network (a single cell) in Figure 4.2. The figure shows two multicast groups: the white MC nodes form the group $\mathcal{G}_1 = \{n_1, n_2, n_3, n_4, n_6\}$ that should receive packet a, and the gray MC nodes form the group $\mathcal{G}_2 = \{n_5, n_7, n_8\}$ that should receive packet b. The set besides each MC represents the channels available to that MC. Node n_0 represents the MR of the cell, and it has all the five channels available (following the CM channel availability model). Table 4.1, summarizes the basic idea of assisted multicast for the network in Figure 4.2. The first two rows show an optimal multicast schedule without any form of assistance, the first of the two shows the transmissions as $(transmitter, packet, channel)$ tuples, and the second shows the receptions as $(receivers, packet, channel)$ tuples. Similar pairs of rows are presented for three levels of assistance, each of which corresponds to exploiting an additional assistance operation. Columns in Table 4.1 correspond to time slots. As the table explains, under no form of assistance, the best the MR can do is 6 time slots. By adding intra-group assistance, i.e., allowing members of the same group to forward packets to each other, the total multicast time was reduced to 5 slots. By extending the assistance to inter-group the total time was further reduced to 4 time slots. Finally, by allowing nodes to exchange coded (bitwise XORed) packets, the total time was reduced to 3 slots.

In the schedule in the last pair of rows (which uses the three levels of assistance), note that MCs n_1 and n_6 (interested in packet a) have received packet b in slot T_1, and MC n_5 (interested in packet b) has received packet a in slot T_2. Therefore, all the three MCs will be able to decode the $a \oplus b$ packet they received in slot T_3 and extract their own data. This schedule is presented in Figure 4.2. It is worth pointing out that scheduling the overhearing opportunities can highly affect the achievable gain.

4.3.2 Definitions

We present, in this subsection, some necessary definitions.

Definition 4.3.1. Codeword*: is a group of packets (could be a single packet) coded (bitwise XORed) into one packet.*

Definition 4.3.2. Assistance operation*: is the process of having one MC forward to another MC in slot t a codeword that the latter can use, possibly with the codewords it has overheard in $[0, t-1]$, to extract the data destined to it. If the two MCs belong to the same group, then this operation is called Intra-group assistance. Otherwise, it is called Inter-group assistance.*

Definition 4.3.3. Codeword exchange*: is the process of allowing the exchange of codewords in the assistance operation.*

Definition 4.3.4. Multicast period*: is the number of time slots needed by the MR to deliver the data packet destined to a multicast group to all the members of that group.*

Definition 4.3.5. Multicast schedule*: is a schedule of the multicast activity over time and frequency. The schedule should determine for each member of a multicast group (including* *the MR) what to transmit/receive (codeword), on what frequency (channel), and at what time (slot). A multicast schedule is feasible iff the following conditions are satisfied.*

1. *Interference constraint: at any slot t, there can be at most one transmission per channel (We assume that all the nodes of a cell (MCs and the MR) are within the interference range of each other).*

2. *Radio constraint: at any slot t, there can be at most one transmission per node (single radio/node).*

3. *Precedence constraint: For an MC to transmit codeword v at time t, it must receive a set of codewords in $[1, t-1]$ sufficient to construct v.*

Then, the assisted multicast scheduling (AMS) problem in CR-WMNs is defined as follows:

Definition 4.3.6. AMS problem in CR-WMNs*: Given M multicast groups $\{\mathcal{G}_{1,i}, \cdots, \mathcal{G}_{M,i}\}$ managed by MR $a_{0,i}$ in cell i, find a feasible multicast schedule within the cell, with both assistance and codeword exchange operations enabled, that results in the minimum multicast period.*

Table 4.2 summarizes all the notations of this chapter.

4.4 Problem Complexity and Formulation

In this section, we study the complexity of the AMS problem and propose two integer linear programs (ILP's) the cases of unassisted multicast and single multicast group with intra-group assistance.

4.4.1 Single multicast group complexity

We first consider the case of a single multicast group in a single cell of the CR-WMN. In such a case, the only possible form of assistance is the intra-group assistance between the members of the multicast group. To understand the complexity of the *"AMS for a single group"* problem, let us study that of the normal, unassisted, multicast scheduling problem as the latter is a special case of the former.

Table 4.2: Summary of Notations

$\mathcal{A}_i$	$\mathcal{A}_i = \{a_{0,i}, \cdots, a_{A_i,i}\}$ is the set of nodes in cell i. $a_{0,i}$ is the MR managing the cell, and $a_{1,i}, \cdots, a_{A_i,i}$ are the MCs of the cell. $A_i = \|\mathcal{A}_i\| - 1$ is the number of MCs in cell i.
$c(u)$	the cell to which node u belongs
$\mathcal{S}$	the set of multicast sessions in the network, $\|\mathcal{S}\| = M$.
$p(j)$	the current packet to be delivered of multicast session j
$\mathcal{G}_{j,i}$	the set of multicast receivers of session j in cell i, thus $\mathcal{G}_{j,i} \subset \mathcal{A}_i$
$\overline{\mathcal{G}}_{j,i}$	the set of multicast receivers of session j in cell i that have received packet $p(j)$
$\mathcal{G}_i = \cup_{j=1}^{\|\mathcal{S}\|} \mathcal{G}_{j,i}$	the set of all multicast receivers of all multicast sessions in cell i
$\mathcal{N}_{j,i}(u)$	the set of neighbors of node u in $\mathcal{G}_{j,i} \cup \{a_{0,i}\} \backslash \{u\}$
$\mathcal{N}_i(u)$	the set of neighbors of node u in $\mathcal{G}_i \backslash \{u\}$
$\mathcal{L}$	the set of licensed channels opportunistically utilized by the CR-WMN, such that $\|\mathcal{L}\| = K$
$\mathcal{L}_j \subseteq \mathcal{L}$	the set of channels available to node j
p_{ON}	the probability of a PU being active, i.e., using its channel
ζ	the interference range
V_u $\mathcal{V}_u$	the set of codewords overheard by MC u the set of all combinations of the codewords in V_u
$\mathcal{X}_i[t]$	the set of multicast transmissions scheduled in cell i in time slot t. Each transmission $x \in \mathcal{X}_i[t]$ is represented by the tuple $(z, v, k, \mathcal{R})$, where z is the transmitter, v is the codeword, k is the channel, and $\mathcal{R}$ is the set of receivers.
$\mathcal{X}_i$	the multicast schedule of cell i such that $\mathcal{X}_i = \{\mathcal{X}_i[1], \cdots, \mathcal{X}_i[t], \cdots, \mathcal{X}_i[T_f]\}$, where T_f is the frame length.

Definition 4.4.1. Unassisted Multicast scheduling for a single group (UMS-Single)*: for a cell i, given a single multicast group $\mathcal{G}_{j,i}$ managed by MR $a_{0,i}$ and the set of available channels for each node $u \in \mathcal{G}_{j,i} \cup \{a_{0,i}\}$. Find a multicast schedule that results in the minimum multicast period given that $a_{0,i}$ is the only transmitter (i.e., no assistance).*

Theorem 4.4.1. *The* UMS-Single *problem is NP-hard.*

Proof. See Appendix A. ☐

As the UMS-Single is a special case of the AMS-Single, the latter is also NP-hard. In other words, any instance of the UMS-Single problem can be mapped into an instance of the AMS-Single problem with all edges between MCs removed (to prevent assistance between MCs). Next, we present two ILP formulations for the two problems. These ILPs will be used in the results section to evaluate the gain of using the assistance operation. Before giving the ILPs, we need to present some notations:

- $T_{j,i}$ is the maximum number of time slots needed to deliver the multicast packet of group $\mathcal{G}_{j,i}$ to all the members of that group, and is given as $T_{j,i} = \min\{|\mathcal{L}_{a_{0,i}}|, |\mathcal{G}_{j,i}|\}$.

- ν^t is a binary variable that is set to 1 if a transmission exists in slot t on any of the channels in $\mathcal{L}$.

- $y^t_{u,k}$ is a binary variable that, if set to 1, means that u transmits on channel k at slot t.

4.4.2 ILP for the UMS-Single problem

The UMS-Single problem for a cell i is formulated as follows, where $\hat{u}$ is the MR of i (i.e., $a_{0,i}$).

ILP-UMS: Minimize $\sum_{t=1}^{T_{j,i}} \nu^t$, subject to:

$$\sum_{k \in \mathcal{L}_{\hat{u}}} y^t_{\hat{u},k} \leq \nu^t, \qquad 1 \leq t \leq T_{j,i} \quad (4.1)$$

$$\sum_{k \in \mathcal{L}_u \cap \mathcal{L}_{\hat{u}}} \sum_{\tau=1}^{T_{j,i}} y^\tau_{\hat{u},k} \geq 1, \qquad u \in \mathcal{G}_{j,i} \quad (4.2)$$

The objective is to minimize the total number of used time slots. Constraint 4.1 guarantees at most one transmission per time slot. Constraint (4.2) guarantees that each MC will receive the data by forcing the MR to transmit on at least one of the channels available to that MC.

4.4.3 ILP for the AMS-Single problem

The ILP presented for the UMS-Single problem can be modified to formulate the AMS problem with intra-group assistance for a single multicast group. We just need to allow MCs to forward the data they receive to their neighbors. The ILP formulation of the AMS-Single problem is as follows:

ILP-AMS: Minimize $\sum_{t=1}^{T_{j,i}} \nu^t$, subject to:

$$\sum_{k \in \mathcal{L}_u} y^t_{u,k} \leq \nu^t, \qquad u \in \mathcal{G}_{j,i} \cup \{a_{0,i}\}, 1 \leq t \leq T_{j,i} \quad (4.3)$$

$$\sum_{k \in \mathcal{L}_u} y^1_{u,k} = 0, \qquad u \in \mathcal{G}_{j,i} \quad (4.4)$$

$$\sum_{k\in\mathcal{L}_u} y^t_{u,k} \leq \sum_{w\in\mathcal{N}_{j,i}(u)} \sum_{k\in\mathcal{L}_u\cap\mathcal{L}_w} \sum_{\tau=1}^{t-1} y^\tau_{w,k}, \quad u\in\mathcal{G}_{j,i}, 2\leq t\leq T_{j,i} \tag{4.5}$$

$$\sum_{w\in\mathcal{N}_{j,i}(u)} \sum_{k\in\mathcal{L}_u\cap\mathcal{L}_w} \sum_{\tau=1}^{T_{j,i}} y^\tau_{w,k} \geq 1, \qquad u \in \mathcal{G}_{j,i} \tag{4.6}$$

$$\sum_{u\in\mathcal{G}_{j,i}\cup\{a_{0,i}\}:k\in\mathcal{L}_u} y^t_{u,k} \leq 1, \qquad k \in \mathcal{L}, 1 \leq t \leq T_{j,i} \tag{4.7}$$

Constraint (4.3) guarantees that at most one transmission per node (MR or MC) exists in a time slot. Using constraint (4.4), we forbid MCs from transmitting in the first time slot as they have not received the multicast packet yet. Constraint (4.5) guarantees that no MC transmits on any channel at slot t before it receives the packet from at least one neighbor, on a channel common between the two, in $[1, t-1]$. We guarantee the delivery of the multicast packet to each MC by constraint (4.6). Constraints (4.7) and (4.3) guarantee one transmission per channel and one transmission per node in each time slot respectively.

4.4.4 The complexity of the AMS problem with multiple multicast groups

Apparently, the AMS problem with multiple multicast groups is at least as hard as the AMS with a single group, which is NP-hard as proved in the previous subsection. In fact, the ILP formulation of the AMS problem with multiple groups is more complicated because of the *codeword exchange operation*. Specifically, an MC cannot transmit a codeword v at time t unless it receives a set of codewords sufficient to construct v. To embed this fact into the ILP, we need to take into consideration all combinations of native multicast packets which will increase the number of variables and constraints exponentially. Moreover, the constraint which ensures that each MC receives its multicast packet is also more complicated. Instead of a unique packet that satisfies the constraint in the case of a single group, a group of decodable codewords can satisfy the delivery constraint in the case of multiple groups with the *codeword exchange operation*. This requires us to take into consideration all the combinations of decodable codewords from which an MC can extract its packet. This will also increase the number of constraints exponentially. Therefore, we do not propose an ILP formulation for the AMS problem with multiple groups.

4.5 Heuristic Solution for the AMS problem

In this section, we propose a heuristic algorithm to solve the AMS problem with multiple multicast groups. The algorithm is greedy-based in the sense that it deals with each slot independently and tries to make the optimal decision at this slot. However, finding this optimal decision in each time slot is not an easy task. In fact, it can be shown that for the case of a single multicast group, scheduling the transmissions of the MR and covered MCs at a time slot t (those which have received the multicast packet in $[1, t-1]$) in a slot t such that the packet is delivered to the maximum number of uncovered MCs is NP-hard (assuming of course that covered MCs may *assist* uncovered ones). Therefore, we divide the scheduling task in *a single time slot* t into three phases.

- **Phase-1**: Scheduling the MR transmission (what codeword to transmit, and on which channel).

- **Phase-2**: Scheduling the assistance operation for each assistance candidate (what codeword to transmit and on which channel). An *assistance candidate* is an MC that was not scheduled to receive data in the first phase, and has received at least one codeword in $[1, t-1]$.

- **Phase-3**: Scheduling overhearing opportunities for overhearing candidates. An overhearing candidate is an MC that was not scheduled as a transmitter (assistant MC) or a receiver in the first two phases. Such an MC has the choice to overhear any of the scheduled codeword transmissions it can. It shall overhear the codeword transmission that has the highest potential of being beneficial to the MC itself or any of its neighbors later.

Note that all these operations are scheduled over frequency channels only and not over time. Before presenting the details of each of the three phases, we provide some terminology.

Let V_u be the set of overheard codewords by MC u up until the current time slot. Also, let $p(j)$ be the current packet to be delivered of multicast session j, and let an MC $u \in \mathcal{G}_{j,c(u)}$ where $c(u)$ is the cell to which u belongs. Assume that V_u does not produce $p(j)$, i.e., no combination of the codewords in V_u can produce $p(j)$. Then, the set of useful codewords to u (those that u can use along with V_u to decode and extract $p(j)$ from) can be determined as follows. Define $\mathcal{V}_u$ as the set of all combinations, bitwise XORs, of the codewords in V_u, i.e., $|\mathcal{V}_u| = 2^{|V_u|} - 1$. Then, $p(j) \oplus \mathcal{V}_u^l$ is a useful codeword for MC u, where $\mathcal{V}_u^l$ denotes the l^{th} combination (codeword) of $\mathcal{V}_u$, for all $1 \leq l \leq |\mathcal{V}_u|$.

Let us consider **Phase-1**, and let $\overline{\mathcal{G}}_{j,i} \subseteq \mathcal{G}_{j,i}$ be the set of MCs in cell i that belong to session j and have received $p(j)$. Then, the best $(codeword, channel)$ schedule for the MR $\hat{u}$ that belongs to cell i in a given time slot is found as follows:

$$(v^*, k^*) = \underset{(v\in\mathcal{V}, k\in\mathcal{L}_{\hat{u}})}{\text{argmax}} \sum_{j\in\mathcal{S}} \sum_{u\in\mathcal{G}_{j,i}\backslash\overline{\mathcal{G}}_{j,i}} \begin{cases} 1 & \text{If } k \in \mathcal{L}_u,\ v{=}p(j) \\ & \text{or } v \oplus p(j) {\in} \mathcal{V}_u \\ 0 & \text{otherwise} \end{cases} \tag{4.8}$$

where $\mathcal{S}$ is the set of multicast sessions, and $\mathcal{V}$ is defined as,

$$\mathcal{V} = \left\{ p(j) \cup \bigcup_{l=1}^{|\mathcal{V}_u|} p(j) \oplus \mathcal{V}_u^l : \forall j \in \mathcal{S},\ u \in \mathcal{G}_{j,i}\backslash\overline{\mathcal{G}}_{j,i} \right\} \tag{4.9}$$

Equation (4.8) finds, for the MR, the $(codeword, channel)$ pair that serves the maximum number of unserved MCs at a particular time slot[1]. The same approach is used for the second phase, namely, scheduling the assistance operation. For an assistance candidate MC u, where $c(u) = i$, the optimal $(codeword, channel)$ in a time slot t is found as follows:

$$(v_u^*, k_u^*) = \underset{(v\in\mathcal{V}_u, k\in\mathcal{L}_u/\mathcal{K}[t])}{\text{argmax}} \sum_{j\in\mathcal{S}} \sum_{w\in\mathcal{G}_{j,i}\backslash\overline{\mathcal{G}}_{j,i}} \begin{cases} 1 & \text{if } k \in \mathcal{L}_w,\ v{=}p(i) \\ & \text{or } v{\oplus}p(j){\in}\mathcal{V}_w, \\ & w{\in}\mathcal{N}_i(u) \\ 0 & \text{otherwise} \end{cases} \tag{4.10}$$

[1]Note that $u \in \mathcal{G}_{j,i}\backslash\overline{\mathcal{G}}_{j,i}$ implies, by definition, that $p(j) \notin \mathcal{V}_u$ in (4.8).

where $\mathcal{K}[t]$ is the set of busy channels in time slot t.

The last phase is to schedule the overhearing operation for MCs that are not receiving data or participating in the assistance operation. The basic idea is for an MC to overhear the codeword that is *useful* to the maximum number of its neighbors. Let $\mathcal{X}_i[t]$ be the set of all transmissions in cell i at time slot t, represented as $(transmitter, codeword, channel, receivers)$ tuples. Let z_l, v_l, k_l, and $\mathcal{R}_l$ denote the transmitter, codeword, channel, and set of receivers of multicast transmission x_l. Then, for an overhearing candidate MC u, where $c(u)$=i, the best transmission x^*=$(z_u^*, v_u^*, k_u^*, \mathcal{R}_u^*)$ to overhear is given as:

$$(z_u^*, v_u^*, k_u^*, \mathcal{R}_u^*) = \operatorname*{argmax}_{(z,v,k,\mathcal{R})\in\mathcal{X}_i[t]} \sum_{j\in\mathcal{S}} \sum_{w\in\mathcal{G}_{j,i}\setminus\overline{\mathcal{G}}_{j,i}} \begin{cases} 1 & \text{if } v\in \bigcup_{l=1}^{|\mathcal{V}_w|} \mathcal{V}_w^l \oplus p(j), \\ & w \in \mathcal{N}_i(u) \\ 0 & \text{otherwise} \end{cases} \tag{4.11}$$

The AMS heuristic approach, denoted HAMS, is outlined in Algorithm 4. The first phase, i.e., scheduling the MR transmission, is expressed by lines $[7-15]$. The phase of scheduling the assistance operation is expressed by lines $[18-32]$. The phase of scheduling overhearing opportunities is expressed in the loop starting at line 33. Finally, unnecessary scheduled overhearings (those which were not used to decode any useful packet) are removed at line 37.

4.6 Collision-free Scheduling across Multiple cells

So far, we have been concerned with scheduling the multicast activity within a single cell. The potential conflict between adjacent cells, due to collisions, was not taken into consideration. In this section, we investigate possible solutions to prevent conflicts between the schedules of adjacent cells and limit the effect of such conflict on the gain achieved by the proposed assistance mechanism. We will investigate two approaches to avoid/resolve conflicts; a *proactive* approach that guarantees conflict-free schedules, and a *reactive* approach that allows conflicts, and then resolves any conflicts after they are detected.

Algorithm 4: HAMS: Heuristic solution for the AMS problem for cell i.

input : Multicast groups $\{\mathcal{G}_{1,i}, \cdots, \mathcal{G}_{M,i}\}$, $\mathcal{L}_i\ \forall i \in \mathcal{G}_i \cup \{a_{0,i}\}$.

$V_u \leftarrow \emptyset\ \forall u \in \mathcal{G}_i,\ \overline{\mathcal{G}}_{j,i} \leftarrow \emptyset\ \forall j \in \mathcal{S},\ t = 0$;

while $\exists j \in \mathcal{S} : |\overline{\mathcal{G}}_{j,i}| < |\mathcal{G}_{j,i}|$ **do**

    $t \leftarrow t + 1$;

    $\mathcal{B}[t] \leftarrow \emptyset$; *//Busy MCs in slot t*

    $\mathcal{X}_i[t] \leftarrow \emptyset$; *//Transmissions in slot t*

    $\mathcal{K}[t] \leftarrow \emptyset$; *//Busy channels in slot t*

    Find the optimal (*codeword*, *channel*) for the MR using eq. (4.8), let it be (v^*, k^*) ;

    $\mathcal{R} \leftarrow \emptyset$;

    forall $(j, u) : j \in \mathcal{S},\ u \in \mathcal{G}_{j,i} \backslash \overline{\mathcal{G}}_{j,i}, k^* \in \mathcal{L}_u,\ v^* \oplus p(j) \in \mathcal{V}_u$ **do**

        $\mathcal{R} \leftarrow \mathcal{R} \cup \{u\}$;

        $V_u \leftarrow V_u \cup \{v^*\}$;

        $\overline{\mathcal{G}}_{j,i} \leftarrow \overline{\mathcal{G}}_{j,i} \cup \{u\}$;

    $\mathcal{B}[t] \leftarrow \mathcal{B}[t] \cup \{a_{0,i}\} \cup \mathcal{R}$;

    $\mathcal{K}[t] \leftarrow \mathcal{K}[t] \cup \{k^*\}$;

    $\mathcal{X}_i[t] \leftarrow \mathcal{X}_i[t] \cup \{(a_{0,i}, v^*, k^*, \mathcal{R})\}$;

    $\mathcal{R} \leftarrow \emptyset$;

    //Schedule the assistance operation

    while $|\mathcal{G}_i \backslash \mathcal{B}[t]| > 2$ **do**

        forall $u \in \mathcal{G}_i \backslash \mathcal{B}[t]$ **do**

            Find the optimal (*codeword*, *channel*) for MC u using eq. (4.10) and let that be (v_u^*, k_u^*), and let the value of the maximum be α_u^*;

        $\hat{u} = \underset{u \in \mathcal{G}_i \backslash \mathcal{B}[t]}{\text{argmax}}\ \alpha_u^*$;

        if $\alpha_{\hat{u}}^* = 0$ **then**

            break;

        else

            $\mathcal{R} \leftarrow \{u : \exists j \in \mathcal{S} \text{ where } u \in \left(\mathcal{G}_{j,i} \backslash \overline{\mathcal{G}}_{j,i}\right) \cap (\mathcal{N}_i(\hat{u}) \backslash \mathcal{B}[t]), v_{\hat{u}}^* \oplus p(j) \in \mathcal{V}_u, k_{\hat{u}}^* \in \mathcal{L}_u\}$;

            $\mathcal{X}_i[t] \leftarrow \mathcal{X}_i[t] \cup \{(\hat{u}, v_{\hat{u}}^*, k_{\hat{u}}^*, \mathcal{R})\}$;

            $\mathcal{B}[t] \leftarrow \mathcal{B}[t] \cup \{\hat{u}\} \cup \mathcal{R}$;

            $\mathcal{K}[t] \leftarrow \mathcal{K}[t] \cup \{k_{\hat{u}}^*\}$;

            forall $(j, w) : j \in \mathcal{S},\ w \in \mathcal{N}_i(\hat{u}) \cap \mathcal{G}_{j,i} \backslash \overline{\mathcal{G}}_{j,i},\ k_{\hat{u}}^* \in \mathcal{L}_w,\ v_{\hat{u}}^* \oplus p(j) \in \mathcal{V}_w$ **do**

                $\overline{\mathcal{G}}_{j,i} \leftarrow \overline{\mathcal{G}}_{j,i} \cup \{w\}$;

                $\mathcal{B}[t] \leftarrow \mathcal{B}[t] \cup \{w\}$;

                $V_w \leftarrow V_w \cup \{v_u^*\}$;

    forall $u \in \mathcal{G}_i \backslash \mathcal{B}[t]$ **do**

        Find the optimal transmission $x^* = (z^*, v^*, k^*, \mathcal{R}^*) \in \mathcal{X}_i$ for MC u to overhear using eq. (4.11);

        $V_u \leftarrow V_u \cup \{v_u^*\}$;

        Add u to the receivers, i.e. $\mathcal{R}^*$, of the multicast transmission.

Remove unused overhearings for all MCs;

4.6.1 Proactive approach

Under this approach, whenever an MR calculates the schedule of the cell it manages, it informs all the adjacent cells about the channels it uses in each time slot. Therefore, any MR that needs to calculate/update the multicast schedule of the cell it manages must refrain from using any channel during a particular slot t that an adjacent cell is using in that slot. In other words, when an MR uses the HAMS algorithm to calculate the schedule, it shall add to the set of busy channels in slot t, i.e., $\mathcal{K}[t]$, all the channels that are used by adjacent cells in slot t. When a particular channel is no longer used in a specific time slot in a cell, the MR managing that cell must inform adjacent cells about this change.

This approach guarantees collision free schedules. Therefore, no post-scheduling phase is needed for resolving collisions. Moreover, it is simple to implement. However, this approach may limit the potential gain of the proposed assistance mechanism because some cells may not be able to utilize the full set of channels available to its nodes (in order to avoid collision with adjacent cells). Furthermore, maintaining up-to-date channel usage information across adjacent cells incurs communication overhead.

This approach is outlined in Algorithm 5. When an MR is ready to activate the schedule of the cell it manages, it sends a scheduling request to all adjacent cells (line 0). This request will help us resolve concurrent activation of schedules which may result in collisions. Each MR that receives a scheduling request from an adjacent MR will reply with a positive acknowledgment if it is not in the process of activating its own schedule in the current frame (line 15). However, if the MR that has received a scheduling request is currently in the process of activating its own schedule, it will reply with a negative (positive) acknowledgment if it has a lower (higher) priority than the MR sending the request (lines 9-14). The MR that receives positive acknowledgments from all of its adjacent MRs activates its schedule in the next frame (line 3-5). It shall also update all adjacent MRs with the channels it uses in each time slot.

Algorithm 5: Proactive Collision-Avoidance (PCA)

if *an MR* $a_{0,i}$ *needs to activate the multicast schedule of cell* i **then**
  It broadcasts a scheduling request packet to all adjacent MRs (those managing adjacent cells);
  if *all adjacent MRs accept the request by sending a positive acknowledgment (+ACK)* **then**
    Activate the schedule in the next frame;
    Inform all adjacent MRs about the used channels in each slot;
  else
    Retry the activation in the next frame
if *MR* $a_{0,i}$ *receives a scheduling request from an adjacent MR* $a_{0,j}$ **then**
  if $a_{0,i}$ *is trying to activate its schedule in the current frame* **then**
    if *cell* i *has a higher priority than cell* j **then**
      $a_{0,i}$ replies with a *negative* acknowledgment to $a_{0,j}$;
    else
      $a_{0,i}$ sends a *positive* acknowledgment to $a_{0,j}$;
      $a_{0,i}$ aborts the schedule activation and retries in the next frame;
  else
    $a_{0,i}$ sends a *positive* acknowledgment to $a_{0,j}$;

4.6.2 Reactive approach

Under this approach, each MR calculates the schedule of the cell it manages without taking adjacent cells into consideration. This means that collisions may occur in some time slots between adjacent cells. Therefore, a collision resolution procedure is needed. The advantage of this approach is that each cell can obtain the full gain of the proposed assistance mechanism. Before discussing the collision resolution procedure, we need to highlight some properties of the schedule that the HAMS algorithm produces.

1. *The precedence property*: the first property is the precedence relationship imposed on transmissions. This relationship resembles the fact that an assistance candidate MC cannot perform its assistance by transmitting a particular codeword unless it has already received it (or a combination that can produce it) through an earlier transmission(s). For example, transmissions $x_1 = \{z_1, v_1, k_1, (z_3, \cdots)\}$ and $x_2 = \{z_2, v_2, k_2, (z_3, \cdots)\}$ must precede transmission $x_3 = \{z_3, v_1 \oplus v_2, k_3, (\cdots)\}$. We represent this precedence relationship

using the following notation, $x_2 \prec x_3$ and $x_1 \prec x_3$. In general, $x_i \prec x_j$ if the codeword v_i was necessary to construct the codeword v_j in the original cell schedule.

2. *The conflict property*: the second property is that any two multicast transmissions x_1 and x_2 cannot be scheduled in the same time slot if any of the following *collision conditions* hold.

   (a) $z_1 = z_2$.

   (b) $\mathcal{R}_1 \cap \mathcal{R}_2 \neq \emptyset$.

   (c) $k_1 = k_2$ and $c(z_1) = c(z_2)$.

   (d) $k_1{=}k_2$, $c(z_1){\neq}c(z_2)$, and $\exists r_1{\in}\mathcal{R}_1 : \|r_1, z_2\|{\leq}\zeta$ or $\exists r_2{\in}\mathcal{R}_2 : \|r_2, z_1\|{\leq}\zeta$ where ζ is the interference range.

   Let $\mathcal{F}(x_i, x_j)$ be the collision function defined as follows:

$$\mathcal{F}(x_i, x_j) = \begin{cases} 1 & \text{if any of the collision conditions} \\ & \quad \text{mentioned earlier is satisfied} \\ 0 & \text{otherwise} \end{cases} \tag{4.12}$$

Given the two properties explained earlier, proposing a *distributed* algorithm that can resolve collisions without wasting the gain achieved by the assistance operation is not an easy task. Therefore, we adopt the *proactive* approach as the solution for collision resolution. Furthermore, we propose an ILP formulation to resolve the collisions for the reactive approach. The performance of this ILP will be used a baseline reference to evaluate the performance of the reactive approach. The basic idea of the ILP is to fit the schedules of C cells in the shortest time frame possible, we refer to this frame as the network span, such that no collisions happen in any time slot. The shorter the network span the better because it leads to higher throughput for the CR-WMN and smaller probability of collision with the primary network.

Given the multicast schedules $\{\mathcal{X}_1, \cdots, \mathcal{X}_C\}$ of the total C cells in the network, such that $\mathcal{X}_i = \{\mathcal{X}_i[1], \cdots, \mathcal{X}_i[\tau_i]\}$ where τ_i is the length of the schedule $\mathcal{X}_i$ which is obtained using Algorithm 4. Let $\overline{\mathcal{X}}_i = \bigcup_{t=1}^{\tau_i} \mathcal{X}_i[t]$. Also, let $\tau_{max} = \max_{1 \leq i \leq C} \tau_i$. Then, the ILP formulation is shown next.

Minimize $\sum_{t=1}^{\tau_{max}} t \cdot \nu^t$, subject to:

$$\omega_{j,n,t} \leq \nu^t, \qquad 1 \leq n \leq C, x_j \in \overline{\mathcal{X}}_c \quad (4.13)$$

$$\omega_{j,n,t} + \omega_{i,m,t} \leq 1, \qquad \begin{matrix} 1 \leq t \leq \tau_{max}, 1 \leq n, m \leq C, \\ x_i \neq x_j, x_i \in \overline{\mathcal{X}}_m, x_j \in \overline{\mathcal{X}}_n, \digamma(x_i, x_j) = 1 \end{matrix} \quad (4.14)$$

$$\omega_{j,n,t} \leq \sum_{\hat{t}=1}^{t-1} \omega_{i,n,\hat{t}}, \quad \begin{matrix} 1 \leq t \leq \tau_{max}, 1 \leq n \leq C, \\ x_i, x_j \in \overline{\mathcal{X}}_n, x_i \prec x_j \end{matrix} \quad (4.15)$$

$$\sum_{t=1}^{\tau_{max}} \omega_{j,n,t} = 1, \qquad 1 \leq n \leq C, x_j \in \overline{\mathcal{X}}_n \quad (4.16)$$

The objective of this ILP is to minimize the length of the network span. Constraint (4.14) guarantees collision free solution. Constraint (4.15), on other hand, maintains the precedence relationship between the multicast transmissions within the same cell. Lastly, constraint (4.16) guarantees that each transmission is scheduled in a time slot.

4.7 Handling transmission failures

In a cognitive radio network, channel availability is not guaranteed for SUs. The presence of a PU in a particular slot will force the SU transmitter which is scheduled to transmit in this slot to abandon its schedule and back off. Therefore, there is a need to devise a mechanism to cope with the potential interruption in the calculated schedules due PU activity. We introduce such a mechanism in this section.

We need to distinguish here between the recovery of an MR and an MC failed transmissions. The HAMS algorithm guarantees that the MR has something to transmit in each slot of the frame. Therefore, if it fails to transmit in a particular slot, there is no way for it to recover that failed transmission without discarding some other scheduled transmission(s). Therefore, when the MR fails to transmit, and the delivery delay exceeds the maximum tolerable delay for the multicast application, rescheduling will be triggered and extra slots will be added to help the MR drain its queues, as it will be explained below.

On the other hand, the MC transmitters might have some transmission opportunities throughout the frame to use for recovery. It would be beneficial to make use of such opportunities not just to deliver the missed transmission earlier, but also to unblock any future transmissions which are pending on the reception of the codeword of the missed transmission. For example, assume that MC z is supposed to transmit codeword $v = p_1 \oplus p_2$ in slot t given that it has received p_1 in slot $t-1$ and p_2 in slot $t-2$. If a PU occupies the channel at $t-2$ causing the scheduled transmission to fail, both codewords p_2 and $p1 \oplus p_2$ will be delayed.

4.7.1 Recovery process

As explained earlier, we need a recovery process to fix the schedule in case of any interruptions caused by PU activity. In this subsection, we propose an online recovery scheduling algorithm that monitors the dynamics of a queueing system maintained at each node and based on which calculates the recovery schedules. Before we propose the recovery algorithm, we need to illustrate the queueing strategy on which the recovery algorithm will rely. Each node, i, that is scheduled to transmit in at least one slot (either the MR or an assistance MC) will maintain the following queues:

- **Input queue (IQ)**: this queue holds the received codewords, and it is parameterized by the frame ID (i.e., a node i will maintain separate IQ's for each frame f, $IQ_i[f]$).

- **Availability queue (AQ)**: this is a virtual queue that holds all combinations of the codewords in IQ and is again parameterized by the frame ID, $AQ_i[f]$. Please note that this is a virtual queue used to simplify the algorithm presentation and it is not a physical queue. It is used to indicate that all the combinations needed to construct a scheduled codeword have been received.

- **Delayed Queue (DQ)**: this is a virtual queue holding all the codewords that a transmitter was unable to transmit because they are not yet available (i.e., not present in AQ

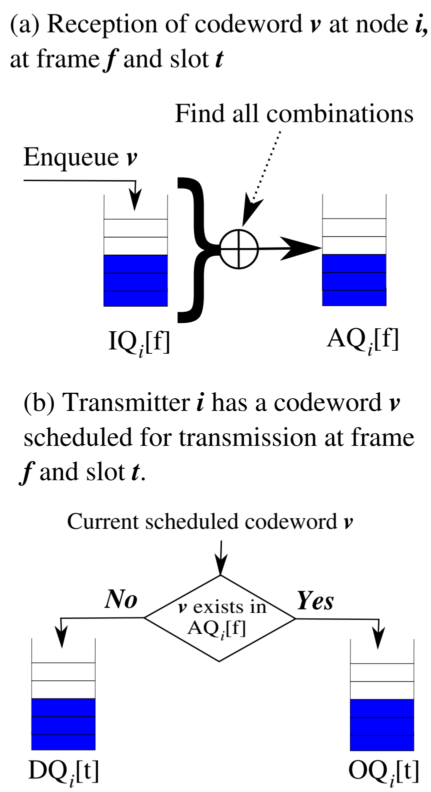


Figure 4.3: Interaction between different queues.

at the time of transmission). This queue is parameterized by the slot ID, $DQ_i[t]$.

- **Output Queue (OQ)**: this is a physical queue holding the codewords that are available (i.e., present in AQ) but the transmitter is unable to transmit because the channel is unavailable at the scheduled transmission time. This queue is parameterized by the slot ID, $OQ_i[t]$.

Figure 4.3 illustrates the interaction between these four queues. Two more points to add:

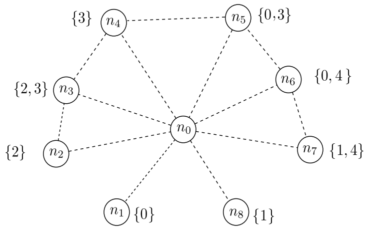
- Whenever queue $AQ_i[f]$ is updated, the following check is performed: $\forall v \in DQ_i[t]$ *($0 \leq t \leq F$): if $v \in AQ_i[f]$, then dequeue v from $DQ_i[t]$ and enqueue it into $OQ_i[t]$,* where F is the frame length.

- Whenever a codeword v is moved out of DQ to the OQ, all codewords in the IQ which are no longer needed to construct a codeword in DQ are removed.

Whenever an MC i fails to use the slot scheduled for it to transmit, say t, it informs the MR via the CCC together with the size of $OQ_i[t]$ (we assume that the MR either has a separate radio for control, or uses time multiplexing). Before the beginning of the next frame, the MR calculates recovery schedules for all nodes with non-zero OQ's giving priority to the ones with the largest OQ size as outlined in Algorithm 6. The MR then informs the MCs about the calculated recovery schedule via the control channel as it will be described later. The extra transmissions scheduled for recovery purposes are valid for one frame only, and the MR will recalculate recovery schedules every frame, as needed.

The MR calculates the recovery schedule for MCs in a greedy manner as outlined in algorithm 6. The OQ with the largest size is processed first. Let that be $OQ_{i^*}[t^*]$. The MR looks up the transmission details (i.e., codeword and receivers) from the original schedule (calculated by HAMS). Then, it iterates over the slots in the frame trying to schedule a transmission that serves the maximum number of receivers and at the same time does not conflict with any scheduled transmission in the original schedule. If it succeeds to serve all the receivers of the failed transmission, it adds the found transmission opportunities to the original schedule. Otherwise, it ignores this OQ (by setting the size to zero to make it ignorable). This operation repeats until there is no more non-zero OQ's to process. The MR then sends the new calculated schedule to all MCs via the control channel (please recall that the extra scheduled transmissions are valid for one frame only). It is also possible that no recovery is possible for some transmitters. In such case, the MR will do nothing. It will just wait for a notification from a receiver MC that the packet delay has exceeded the maximum tolerable delay by the multicast application. If the MR receives such a notification, it triggers the full rescheduling (i.e., running the HAMS again) taking into consideration the avoidance of the channels which have caused consistent growth in OQ's. The MR will also add the minimum number of extra slots that the transmitters (including the MR itself) which have non-zero OQ's can share (using Algorithm 6) to drain their OQ's. If such extra slots are added, the MR will have some idle slots to use for recovering its own OQ's. Once all transmitters drain their OQ's, the MR will shrink the schedule back by removing the extra slots it added earlier, and notify the MCs about this change. Any changes made to the schedule are communicated with the adjacent cells (via the CCC) to maintain the collision free atmosphere.

To explain the recovery behavior, we present the example summarized in Figure 4.4. We simulate the schedule in Figure 4.4 with packet size of 1555 bytes and slot length of 2.43 ms for six seconds. The simulation results are presented in Figure 4.5. In period $[1, 2]$, we blocked channel 3 in slot 3 (i.e., used by a PU) only, and at the same time we disabled failure recovery. Thus, the size of $OQ_4[3]$ increased linearly until it hit 58. Then we enabled the failure recovery in $[2-3]$, while keeping channel 3 busy in slot 3 during this period. According to the schedule, node n_4 can make use of channel 3 in slots $\{0, 1, 5, 6\}$ for recovery. Therefore, the MR scheduled those slots to be used by n_4 to drain $OQ_4[3]$, and the queue quickly drained at rate 4 $packets/frame$. In period $[3-4]$, we blocked channel 3 in slot 2 and disabled failure recovery, and therefore $OQ_5[2]$ has built up. We enabled the recovery back at time $t = 4$, and blocked channel 3 in slots 0 and 1 till the end to the simulation period. According to the schedule, node n_5 can make use of channel 3 in slots 5 and 6. Therefore, the MR schedules those slots for n_5 to use. While $OQ_5[2]$ is draining at rate 2 $packets/frame$, $OQ_4[3]$ is building up because it now has an input rate that is higher than its output rate and at the same time is unable to win any recovery slots because the size of $OQ_5[2]$ is still higher. When the sizes of $OQ_4[3]$ and $OQ_5[2]$ became equal, the MR started to make slots 5 and 6 shareable between n_4 and n_5 for recovery and the recovery rate became 1 packet/frame/node until time 4.7. At $t = 4.7$, we blocked channel 3 in slot 5 leaving only one slot available for recovery purposed. Therefore, the recovery rate dropped down to 1 packet/frame until the OQ's of nodes n_4 and n_5 completely drained at $t = 5.4$. Also, note that the size of $DQ_4[3]$ matches that of $OQ_5[2]$ as expected.



Slot	Transmitter	Channel	Codeword	Receivers
0	n_0	0	a	$\{n_1, n_5, n_6\}$
1	n_0	0	b	$\{n_1, n_5, n_6\}$
2	n_5	3	a	$\{n_4\}$
2	n_0	1	a	$\{n_7, n_8\}$
3	n_4	3	a	$\{n_3\}$
3	n_0	1	b	$\{n_7, n_8\}$
4	n_0	3	$a \oplus b$	$\{n_3, n_4\}$
5	n_0	2	a	$\{n_2\}$
6	n_0	2	b	$\{n_2\}$

Figure 4.4: A case study to illustrate the recovery processes. The figure to the left shows the network topology and channel availability, while the table to the right shows the calculated schedule.

Algorithm 6: Greedy approach to recalculate recovery schedules

input : $OQ_i[t],\ \forall i \in \mathcal{Z},\ 0 \le t < F$;
$\mathcal{Z}$: the set of transmitters in the permanent schedule of the next frame;
F: the frame length;
output: $\mathcal{X}_{total}$: The schedule of the next frame including temporary recovery transmissions;

Copy the permanent schedule of the next frame into $\mathcal{X}_{total}$;
while $\exists(i,t) : OQ_i[t] > 0$ **do**
  Let $OQ_{i^*}[t^*]$ be the OQ with the largest size;
  From the permanent schedule, look up the scheduled transmission at slot t^*. Let that be $x^* = \{i^*, v, k, \mathcal{R}\}$;
  $\mathcal{X} \leftarrow \emptyset$;
  for $\tau = 0; \tau < F; \tau++$ **do**
    if $|\mathcal{R}| = 0$ **then**
      break;
    if *i^* and at least one receiver in $\mathcal{R}$ are idle in $\mathcal{X}_{total}[\tau]$ and shares a common idle channel* **then**
      Let $k^* \in \mathcal{L}_{i^*}$ be an idle channel in τ that is available to the maximum number of receivers represented by the set $\mathcal{R}^* \subseteq \mathcal{R}$;
      $\mathcal{X} \leftarrow \mathcal{X} \cup (i^*, v, k^*, \mathcal{R}^*)$;
      $\mathcal{R} \leftarrow \mathcal{R} \backslash \mathcal{R}^*$;
  if $|\mathcal{R}^*| < |\mathcal{R}|$ **then**
    $OQ_{i^*}[t] = 0$
  else
    $\mathcal{X}_{total} \leftarrow \mathcal{X}_{total} \cup \{x\},\ \forall x \in \mathcal{X}$
return $\mathcal{X}_{total}$;

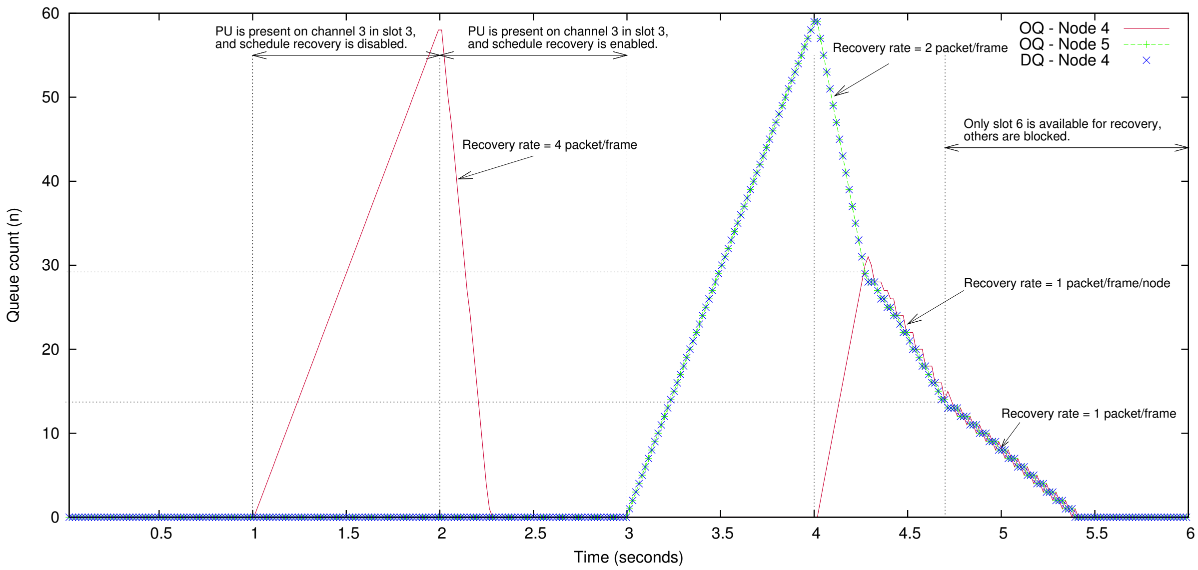


Figure 4.5: The queue dynamics of the case-study summarized in Figure 4.4.

4.8 Performance Evaluation

In this section, we evaluate the performance of the proposed assistance and collision resolution mechanisms. Before presenting the results, we illustrate the channel availability models we used in our simulations. In this section, M will denote the number of multicast sessions, i.e., $M = |\mathcal{S}|$.

4.8.1 Channel Availability Models

In this subsection, we outline a number of possible channel availability models in CRNs. However, we only use the third, i.e., CM, model for our simulations.

1. *The spatial model (SM)*: under this model, a number of PUs, N_{PU}, are uniformly distributed in the network field. Each PU is assigned one frequency channel selected uniformly at random from a pool of K channels. An SU j can use a frequency channel k *iff* all PUs which are assigned channel k are at least at a distance R_p away from j.

2. *The hybrid model (HM)*: as in the spatial model, a number of PUs, N_{PU}, are uniformly distributed in the network field. Each PU is assigned one frequency channel selected

uniformly at random from a pool of K channels. A PU is active (i.e., using the assigned channel) with probability p_{ON}, and inactive with probability $1 - p_{ON}$. An SU j can use a frequency channel k *iff* all *active* PUs which are assigned channel k are at least R_p away from j.

3. *The coexistence model (CM)*: each MR serves both primary and secondary clients, which means that MRs can utilize all frequency channels. SUs are treated as second class users which receive best effort service. Therefore, the set of available channels at SUs will depend on many different factors including, spatial distribution of PUs and SUs, primary traffic loads, load balancing between frequency channels, pricing function (in case of spectrum auction), etc. To keep the model simple, we make all channels available to all MRs, while a channel is made available to an MC with probability P_a.

4.8.2 The gain of receiver assistance

To evaluate the gain of the proposed assistance mechanism, we study a single cell with the number of MCs varying from 5 to 50. The MCs are distributed uniformly at random in a square area of $500m \times 500m$ around the MR which is located in the center of the square area. All nodes (MCs and the MR) are assumed to have the same communication radius of $\sqrt{2} \times \frac{500}{2} = 353.55m$ over all channels. We vary the number of multicast groups M between 1, 3, 4, and 5. Each MC is assigned to any of the M groups uniformly at random, i.e., each MC belongs to exactly one multicast group. Lastly, we have the number of channels $K = 6$ in all experiments. Available channels are determined at each node (MC or MR) according to the CM model.

Intra-group assistance Figure 4.6 shows the gain of using intra-group assistance in a single multicast group. The gain is defined as the percentage reduction in the multicast period of the unassisted multicast achieved by using assisted multicast ($\frac{unassisted - assisted}{unassisted} \times 100\%$). The optimal solutions for the two cases of unassisted multicast and intra-group assisted multicast were obtained using the two ILPs proposed in Section 4.4. We also evaluated the gain of intra-group assisted multicast by scheduling the problem using the HAMS algorithm. Each point is the figure is the average over a 100 randomly generated topologies. As the figure shows, the intra-group assistance achieves a significant gain over the unassisted case that increases with increasing the group size. On the other hand, the HAMS algorithm is performing well by achieving a considerable gain and being always within, on average, one time slot of the optimal solution obtained by ILP formulation of the AMS-Single problem. In fact, HAMS was, on average, ≈ 0.63 slots higher than the optimal assisted multicast schedule, and ≈ 2.11 slots less than the optimal unassisted multicast schedule.

Inter-group assistance We now evaluate the benefit of using each of the three assistance operations: *intra-group assistance*, *inter-group assistance*, and the *codeword exchange operation* for multiple multicast groups. We vary the number of groups M between 3, 4 and 5. For each case, we evaluate the gain using *intra-group assistance only*, *intra- and inter-group assistance*, and *intra- and inter-group assistance with network coding*. For the unassisted multicast case, we find the optimal schedule for each one of the M groups and summing up the optimal multicast periods for all individual groups to obtain the total multicast period. As for the assisted multicast scheduling, we used the HAMS algorithm. Figures 4.7, 4.8, and 4.9 correspond to the cases of $M = 3,\ 4$, and 5 respectively with each point in the figure be the average over a 100 randomly generated topologies. As the figures show, each level of assistance achieves some extra gain in the total multicast period. However, it is apparent that inter-group assistance has more influence on the total gain than the codeword exchange operation, yet the codeword exchange operation can still improve the scheduling performance. Figure 4.10 shows the actual averages of the multicast period for the data presented in Figures 4.7, 4.8, and 4.9.

The effect of channel availability To understand the effect of channel availability on the achievable gain of the assisted multicast, we varied P_a from 0.1 to 0.7 for the cases of $M = 1$ and $M = 5$ as shown in Figures 4.11 and 4.12 respectively. The number of MCs in a single cell is varied between 10, 30, and 50. All MCs are assumed to be members of all multicast groups to nullify the effect of diversity in group membership on the achievable gain. Each point on the curve of any of the two figures is the average of 200 randomly generated instances. As the two figures show, the gain increases as P_a increase until reaching a peak and then starts decreasing. The P_a at which the gain is maximized offers the highest level of diversity in the network, the basic property on which the proposed assistance mechanism relies. Another thing to note from these two figures is that the gain is higher with higher values of M.

4.8.3 Proactive vs. reactive collision resolution

In this section, we study a network of four cells, all of which have the same number of MCs and share the same pool of channels. The number of channels is chosen from the set $\{4, 6, 8, 10, 12\}$, while the number of multicast groups is set to 3. All MCs are members of all multicast groups to nullify the effect of diversity in group membership on the achievable gain. The number of MCs in each cell is chosen from the set $\{20, 50\}$. In each experiment, all the cells have the same number of MCs. The number of cells in the network is 9, arranged in a grid of 3×3 in a field of area $500m \times 500m$. The communication radius for all nodes is $\sqrt{2} \times \frac{500}{6}$, and ζ is twice the communication radius.

Each cell calculates its multicast schedule using the HAMS algorithm. Then, we use the proactive (the PCA algorithm) and the reactive (the ILP) to resolve collisions between adjacent cells. Figure 4.13 shows the performance of the two approaches represented by the ratio of the proactive approach to the reactive approach for both 20 and 50 MCs in each cell. Each point on the curve is the average of 100 randomly generated instances. As the figure indicates, the performance of the proactive approach is close to that of the reactive approach with optimal collision resolution (less than %5 difference). The figure also shows that for small number of channels (4 for example), the proactive performs better than the reactive approach. Therefore, given the simplicity of implementing the proactive approach (compared to the reactive), and the good performance the figure implies, we adopt the proactive PCA algorithm as the collision resolution procedure.

4.9 Chapter Summary

In this chapter, we studied the problem of assisted multicast scheduling in CR-WMNs. We proposed an assistance paradigm that relies on receiver nodes to forward the multicast data to other receivers that have not yet received their own data. Furthermore, network coding was also proposed as another assistance technique that further reduced the total multicast period. Results show that the proposed assistance paradigm achieves a significant gain in reducing the total multicast period, i.e., overall throughput. A proactive collision resolution procedure was also proposed to allow collision-free schedules across multiple cells in a CR-WMN.

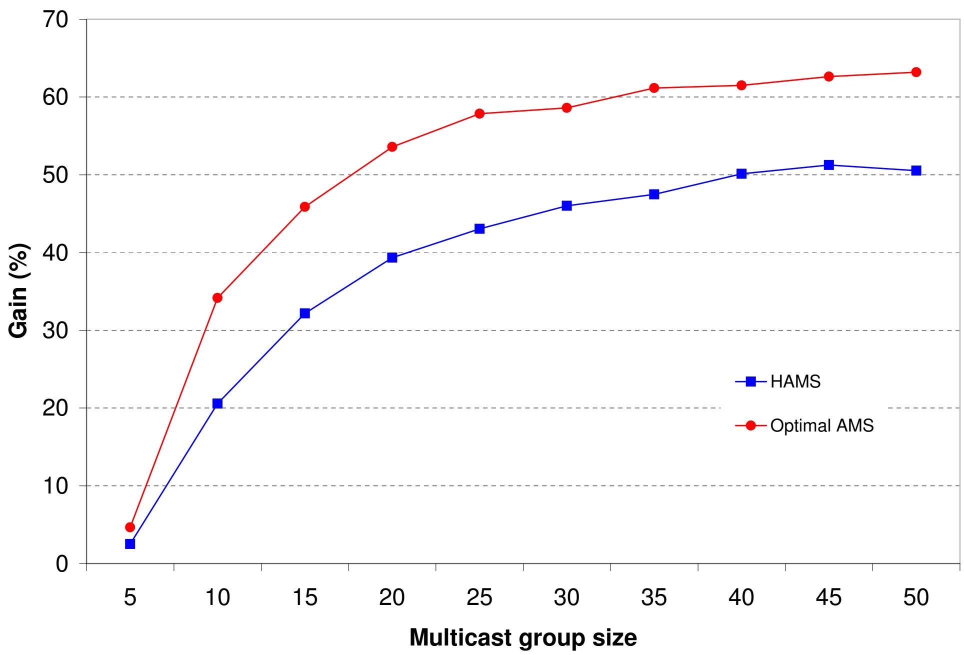


Figure 4.6: The gain of using intra-group assistance in a single multicast group ($Pa = 0.25$).

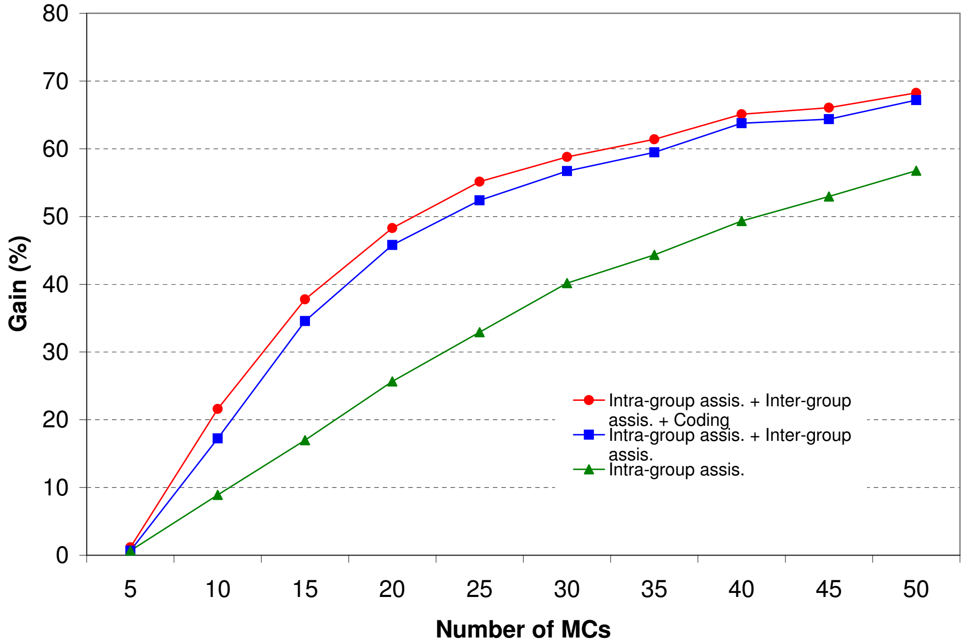


Figure 4.7: Average gain of assisted multicast using different levels of assistance ($M = 3, Pa = 0.25$).

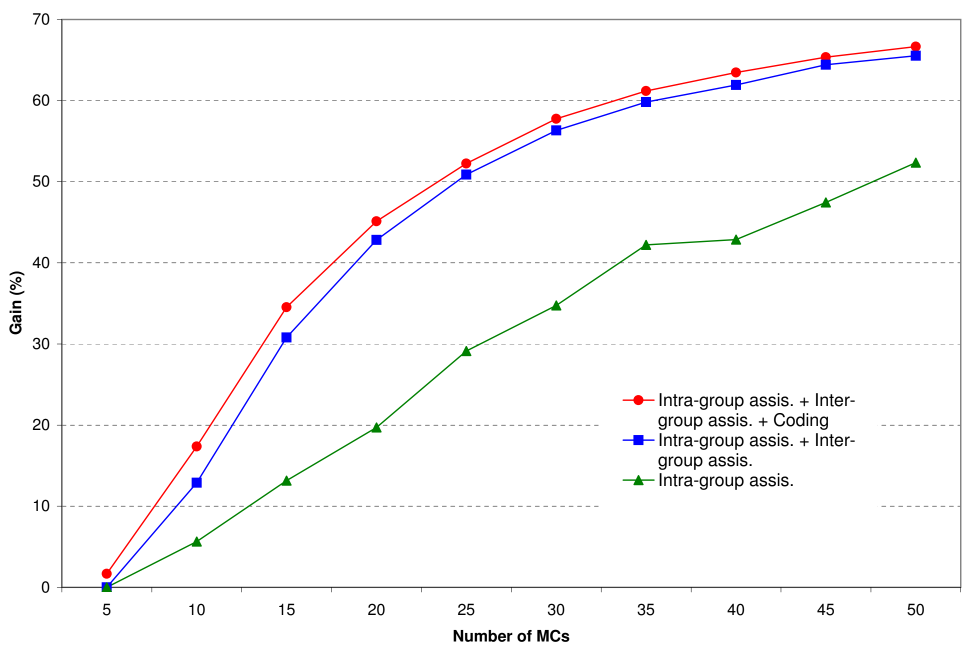


Figure 4.8: Average gain of assisted multicast using different levels of assistance ($M = 4, Pa = 0.25$).

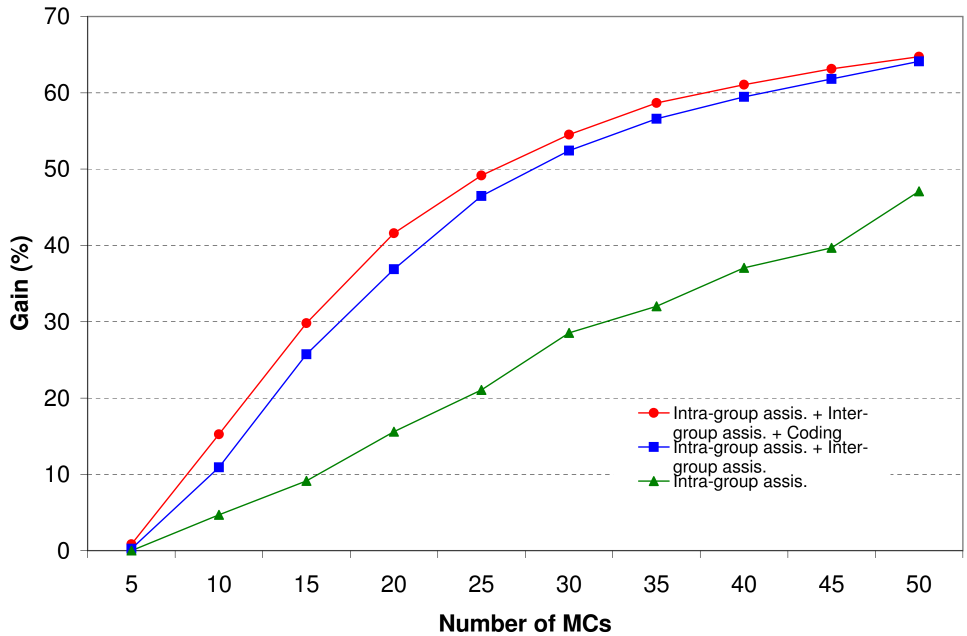


Figure 4.9: Average gain of assisted multicast using different levels of assistance ($M = 5, Pa = 0.25$).

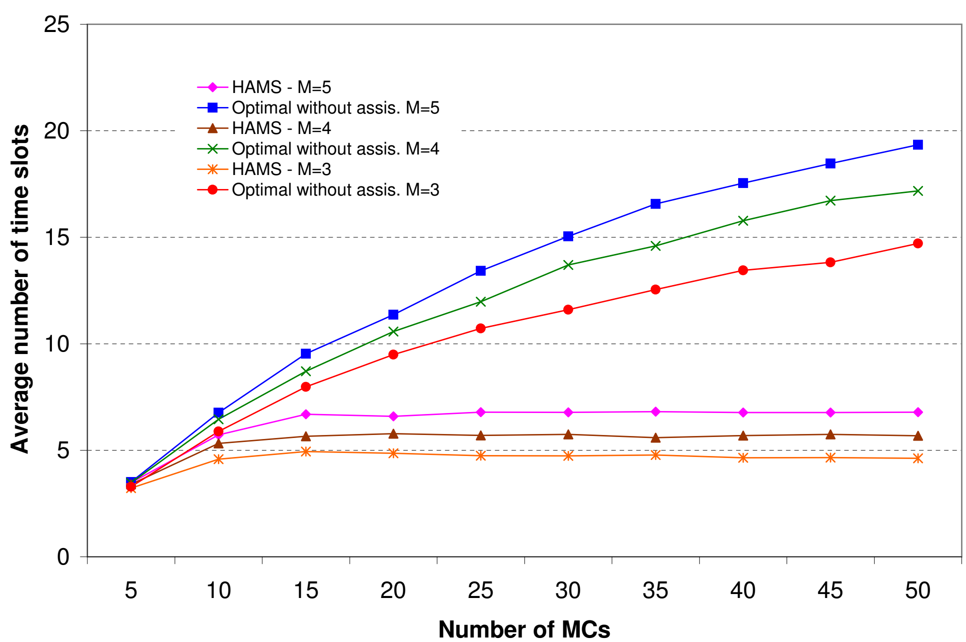


Figure 4.10: Average multicast period with- and without-assistance (M=3, 4, 5, Pa=0.25).

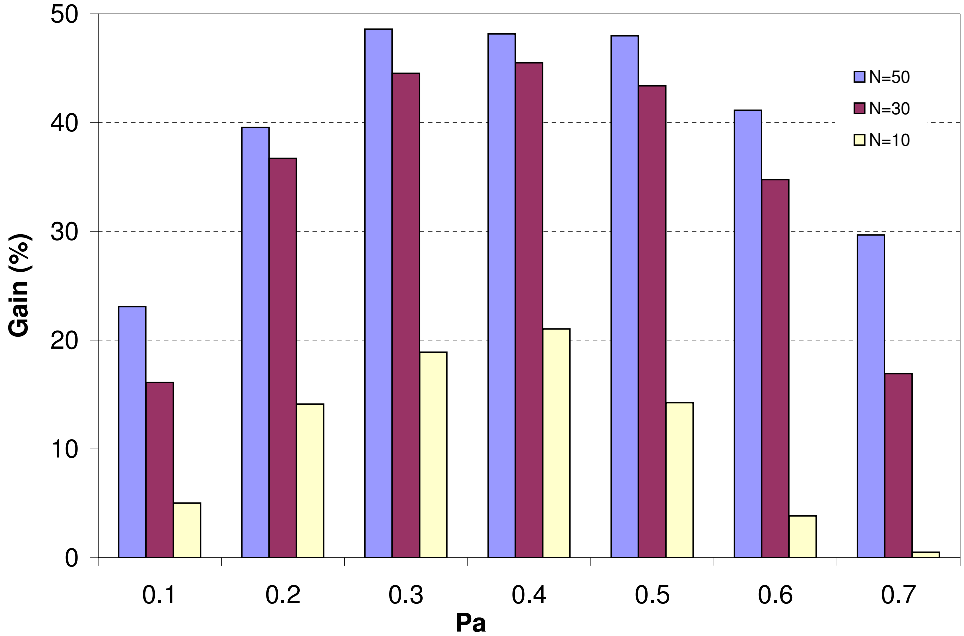


Figure 4.11: The effect of channel availability on the gain of assisted multicast (M=1).

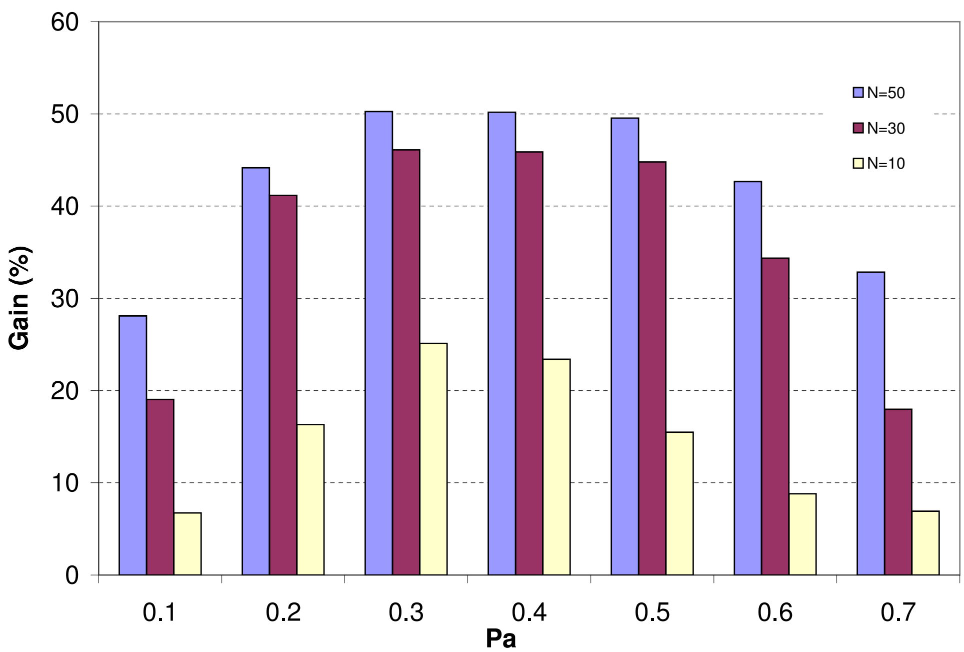


Figure 4.12: The effect of channel availability on the gain of assisted multicast (M=5).

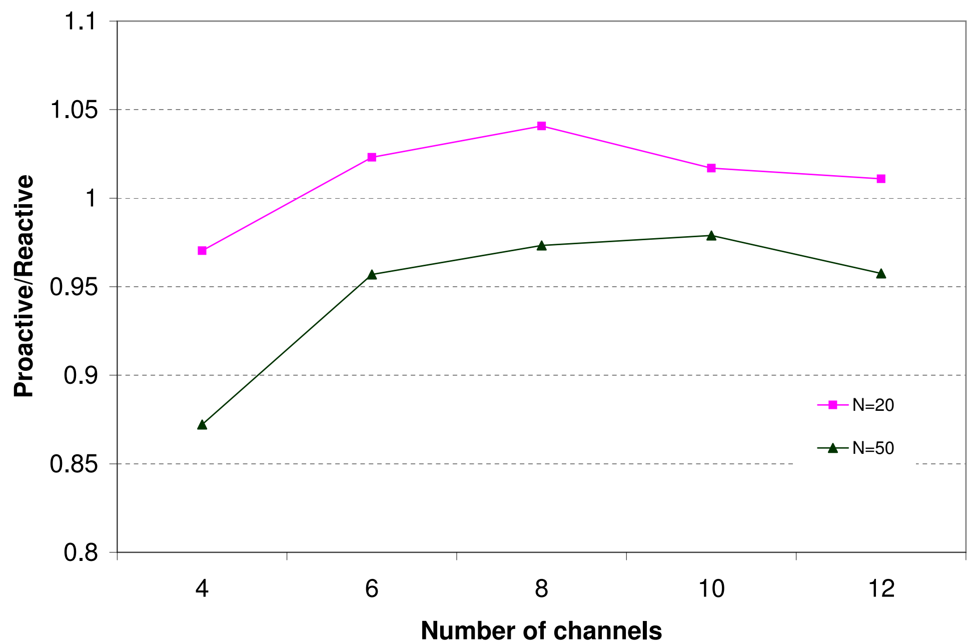


Figure 4.13: Proactive versus reactive collision resolution ($Pa = 0.25$).

CHAPTER 5. ON-DEMAND MULTICAST ROUTING

The routing problem in cognitive radio networks faces unique challenges which are not present in traditional wireless networks. Under the open spectrum model, it may no longer be true that all frequency channels utilized by the network share the same characteristics, like bandwidth and achievable throughput for example. Depending on how the primary (licensed) network utilizes and shares its spectrum resources with the secondary (unlicensed) network, channel availability might be different at different locations and times, and therefore the nodes that belong to the secondary network will observe heterogeneous sets of available channels. Given these facts, there is a need for novel routing algorithms for cognitive radio networks that can optimize the route selection taking into consideration the specific challenges imposed by the coexistence between licensed and unlicensed wireless users.

In this chapter, we address multicast routing and introduce an on-demand multicast routing and channel allocation algorithm that takes the heterogeneity in channel characteristics and availability into consideration. The algorithm is based on dynamic programming and aims at establishing the minimum-cost (maximum-revenue) route to the multicast tree for new members joining a multicast session. The algorithm is general and can be used with different routing objectives.

5.1 Introduction

The multicast routing problem in multihop cognitive radio networks faces some major challenges that are not present in traditional wireless networks. These challenges are mainly caused by the heterogeneity in channel characteristics, and heterogeneity in channel availability among different Secondary Users (SUs). Some examples of these heterogeneities are summarized next:

1. Channels can have different bandwidths and maximum allowed transmission powers. Such constrains may be set by spectrum regulators (like the constraints imposed on the maximum transmission power by FCC). Another possible scenario is for the operators of the licensed network to enforce restrictions on the amount of bandwidth they want to share out of a given band, and the limits on the transmission power on that shared portion of the spectrum to maintain some performance objectives. With such restrictions, different channels can achieve different throughputs.

2. Another form of heterogeneity is that of the channel availability at different SUs. This can be due to the fact that SUs may observe different sets of available (idle) channels depending on the activity of the Primary Users (PUs) in their vicinity. The heterogeneity in channel availability leads to the following unique issues of cognitive radio networks (assuming a single radio interface):

   (a) *Broadcast deformation*: when an SU has neighbors that do not (all) share a common channel with this SU, it cannot broadcast a data unit to all neighboring SUs in one transmission. Therefore, a broadcast might become a number of multicast transmissions, or in the worst case a number of unicast transmissions. This significantly reduces network capacity and increases end-to-end delay.

   (b) *Switching delay*: another source of capacity wastage and delay increase is channel switching [21]. Assume that SU i receives from SU j and forwards the data to SU k. If i cannot find a common channel with j and k together, then it has to use two different channels for transmission over the two links $j \rightarrow i$ and $i \rightarrow k$. Depending on spectral separation between the two channels, the switching delay could be significant. The problem worsens when an SU has to receive from and/or transmit to multiple SUs.

3. From the perspective of SUs, the link reliability can be measured as the probability of a PU interrupting an ongoing SU transmission. This can be heterogeneous across channels and geographical locations depending on the geographical and spectral distribution of

PUs as well as the dynamism of their activities.

Given the examples above, traditional routing algorithms, like shortest path routing, are not suitable for cognitive radio networks and new routing strategies are needed. We propose in this chapter an on-demand dynamic-programming based multicast routing and channel allocation algorithm for wireless cognitive radio mesh networks under the assumption of a single radio interface. The proposed algorithm is generalized in terms of the routing metric, i.e., the same algorithm can be used to find the best route for different routing metrics.

The rest of this chapter is organized as follows. In Section 5.2, we layout the system model and the assumptions. The motivation of this work and the problem formulation are presented in Section 5.3. An optimal channel allocation algorithm along a single path is proposed in Section 5.4. The multicast routing and channel allocation algorithm (OMRA) is then proposed in Section 5.5. The extendability of the proposed routing algorithm to other routing metrics is discussed in Section 5.6. We address the issue of rerouting upon link failures, due to PU activity on the allocated channels, in Section 5.7. In Section 5.8, we evaluate the performance of the proposed OMRA algorithm. We conclude the chapter in Section 5.9.

5.2 System Model

In this section, we layout the system model and assumptions. We consider a wireless cognitive mesh network that consists of a number of mesh routers (MRs), each of which manages a set of mesh clients (MCs), and a single gateway that connects the network to the Internet. Any MR can reach the gateway either directly or through multiple hops of MRs. We are only concerned with multicast traffic that originates from the Internet (like watching TV broadcasts or some special events, like online games) and passes through the gateway to be finally received by SUs, which are members of the multicast session, in the cognitive mesh network. Therefore, we treat the gateway as the (hypothetical) source of all multicast traffic.

We assume the existence of a spectrum sensing and management entity (SSME) that provides an SU (MR or MC) with a list of channels which can be used by that specific SU. In addition to the straightforward implementation of having each SU perform the role of the

SSME itself (or cooperatively with neighboring SUs), a number of other implementations have been proposed in literature. One of these implementations is to have a wireless sensor network infrastructure that is specifically designed to achieve accurate spectrum sensing and provide SUs with information about spectrum occupancy [85]. Spectrum auction is another implementation of an SSME in which some spectrum leasing entity (could be PUs themselves) leases the spectrum to SUs for a specified amount of time [86, 87]. Whatever the adopted implementation of the SSME is, we always assume that each SU has a list of available channels at any point in time. We further assume the existence of a common control channel (CCC) between neighboring SUs. The proposed solutions in this chapter work with any implementation of the CCC, like an actual fixed frequency channel [44], or a virtual CCC [45].

5.3 Motivation and Problem Definition

We will use the end-to-end delay routing metric as a case study to explain the motivation and the algorithm details. In Section 5.6, we will show how the proposed routing algorithm can be used with different routing metrics. Until then, all the discussions and analysis will be under the assumption that end-to-end delay is the routing metric.

To explain the motivation behind this work, consider the example shown in Figure 5.1. The set besides each SU in Figure 5.1.(a) represents the list of channels available to that SU. Two different channel assignments are presented in Figures 5.1.(b) and 5.1.(c) to explain the effect of channel assignment on the throughput and end-to-end delay of multicast traffic. Let us investigate the total time that the forwarding node f needs to relay the multicast data to the multicast receiver-nodes r_1 and r_2, after receiving it from the multicast source s. Assume that the switching delay between two channels operating at central frequencies f_1 and f_2 is a linear function denoted by $d_{sw}(f_1, f_2)$. Let $d_{sw}(f_1, f_2)=\alpha|f_1-f_2|$, where α is the tuning speed (in *seconds*/*Hz*) of the spectrum processor[1]. Also, let $L_n^{i,j}$ be the packet transmission time from MR i to MR j on channel n. This will depend on the packet size and *achievable* channel bit-rate (i.e., Shannon capacity), which in turn depends on the transmission power, the channel

[1]For example, the suggested value of α for the TCI 715 spectrum monitoring system is $1ms/10MHz$ [88].

bandwidth, the coding and modulation schemes, and the link quality between i and j.

To keep the example simple, assume that a packet transmission on any channel takes the same time of L, i.e., $L_n^{i,j} = L \; \forall i, j, n$. Then, the total relay time at node f for case in Figure 5.1.(b) is $L+\alpha$; α (in *seconds*) to switch from channel 3 to channel 4, and L for one transmission on channel 4 to r_1 and r_2. For the case in Figure 5.1.(c), on the other hand, the total relay time is $2L + 4\alpha$ calculated as follows:

$$\text{total relay time} = d_{sw}(1,4) + L + d_{sw}(4,5) + L = 2L + 4\alpha$$

Let τ_b and τ_c denote the throughput of node f in case b and c respectively. Then, $\tau_b = \frac{1}{(L)+L+\alpha+(\alpha)} = \frac{1}{2L+2\alpha}$, and $\tau_c = \frac{1}{(L)+2L+4\alpha+(4\alpha)} = \frac{1}{3L+8\alpha}$. The term (L) in the dominator of both of the formulas above represent the time to receive the packet from s, and the terms (α) and (4α) represent the switching time from the last channel used for transmission back to the channel used for reception for cases (b) and (c) respectively. For 10 $Mbps$ channel rate, 1500 byte packet size, and $1ms$ baseband switching delay, the ratio τ_c/τ_b evaluates to $\approx 38\%$, which is a significant reduction of f's throughput.

On the other hand, the end-to-end delay is also affected. For example, in Figure 5.1, it will take the multicast data $2L + \alpha$ to reach r_1 and r_2 in case (b). However, the end-to-end delay is $2L + 3\alpha$ to r_1 and $3L + 4\alpha$ to r_2 in case (c). Based on the practical values used before, the ratio of the end-to-end in case (c) to case (b) is 158.82% for r_1 and 223.53% for r_2.

Based on these motivational examples, it can be clearly realized that an on-demand solution is needed for such a network because the delay and throughput measures depend on the current situation of the MRs along a candidate route in terms of (1) *the channel availability*, (2) *the number of flows an MR is currently serving*, and (3) *the channels on which these flows are served*. The purpose of this study is to propose such a solution. Without loss of generality, we assume that a single gateway exists in the network from which all multicast sessions originate. Furthermore, we only consider data multicast between MRs as the problem of transmitting the multicast data from an MR to its MC was studied by the authors in chapter 4.

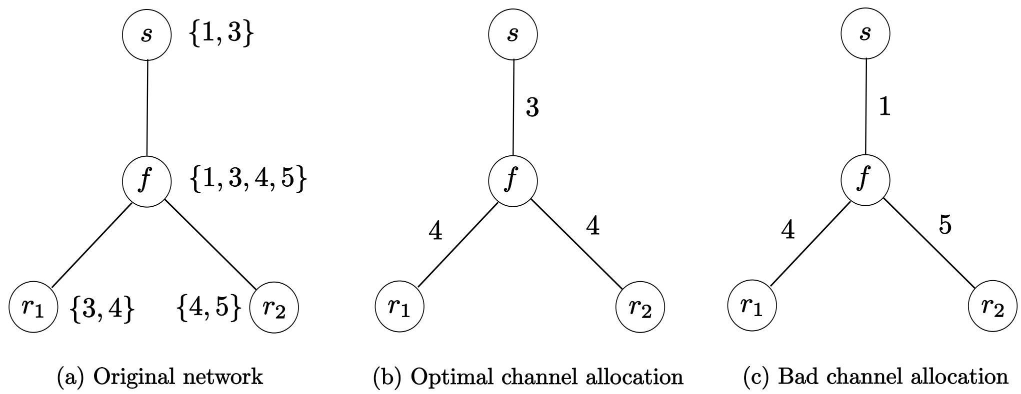


Figure 5.1: An example that illustrates the effect of channel assignment on the throughput and end-to-end delay of multicast traffic.

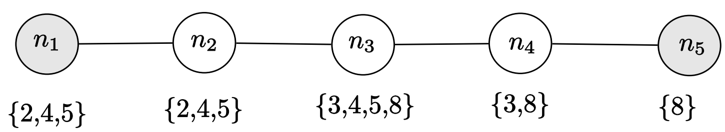


Figure 5.2: A toy example of a path of five MRs from n_1 to n_5, lists under the nodes represent available channels at each node.

5.4 Optimal Channel Assignment on a Route

In this section, we describe the channel assignment problem assuming that the routing problem has been solved. The problem of optimal channel assignment is therefore solved in terms of end-to-end delay along a single path. Then, we will extend the solution proposed for this case to include multicast routing. Consider the path shown in Figure 5.2 under the assumption that all channels have the same rate and bandwidth (i.e. the same packet transmission time L) and using the frequency-switching delay function $d_{sw}(f_1, f_2)$ defined earlier. By investigating all possible channel assignments, it can be easily shown that the channel assignment $\{5, 5, 8, 8\}$ for the links $\{n_1 \rightarrow n_2, n_2 \rightarrow n_3, n_3 \rightarrow n_4, n_4 \rightarrow n_5\}$, respectively, is the optimal with an end-to-end delay of $4L + 3\alpha$, and the assignment $\{2, 5, 3, 8\}$ is the worst with an end-to-end delay of $4L + 10\alpha$. To find the optimal channel assignment on a route, a dynamic program is proposed next.

Table 5.1: All possible channel assignments and their end-to-end delays

$(n_1 \rightarrow n_2)$	$(n_2 \rightarrow n_3)$	$(n_3 \rightarrow n_4)$	$(n_4 \rightarrow n_5)$	Cost
2	4	3	8	$4L + 8\alpha$
2	4	8	8	$4L + 6\alpha$
2	5	3	8	$4L + 10\alpha$
2	5	8	8	$4L + 6\alpha$
4	4	3	8	$4L + 6\alpha$
4	4	8	8	$4L + 4\alpha$
4	5	3	8	$4L + 8\alpha$
4	5	8	8	$4L + 4\alpha$
5	4	3	8	$4L + 7\alpha$
5	4	8	8	$4L + 5\alpha$
5	5	3	8	$4L + 7\alpha$
5	5	8	8	$4L + 3\alpha$

5.4.1 Dynamic programming approach for channel assignment

Given a route (or path) $\mathcal{R}$ that consists of $|\mathcal{R}|$ MRs, numbered from 1 to $|\mathcal{R}|$, and the data flows from MR $|\mathcal{R}|$ to MR 1, i.e, MR $|\mathcal{R}|$ is the source and MR 1 is the destination. The objective is to allocate a channel to each link along $\mathcal{R}$ such that the end-to-end delay is minimized. A formal definition of a dynamic program consisting of *stages*, *states*, *transitions*, and *transition cost* is now developed. MRs along a path are mapped into stages, available channels to MRs into states, and channel assignments into transitions. Let,

- $\mathcal{L}_i$ be the set of channels available at MR i.

- $C_{n,m}^{i-1,i}$ be the cost of making a transition from state n at stage $i-1$ to state m at stage i. For a single path, this cost is given as $C_{n,m}^{i-1,i}=d_{sw}(n,m)+L_n^{i,i-1}$. The cost will be slightly different for multicast routing (see Section 5.5).

- $f^*(i,m)$ be the delay under the optimal solution starting from state m at stage i and ending at stage 1.

- $f^*(|\mathcal{R}|)$ be the minimum end-to-end delay under the optimal solution (channel allocation) for path $\mathcal{R}$.

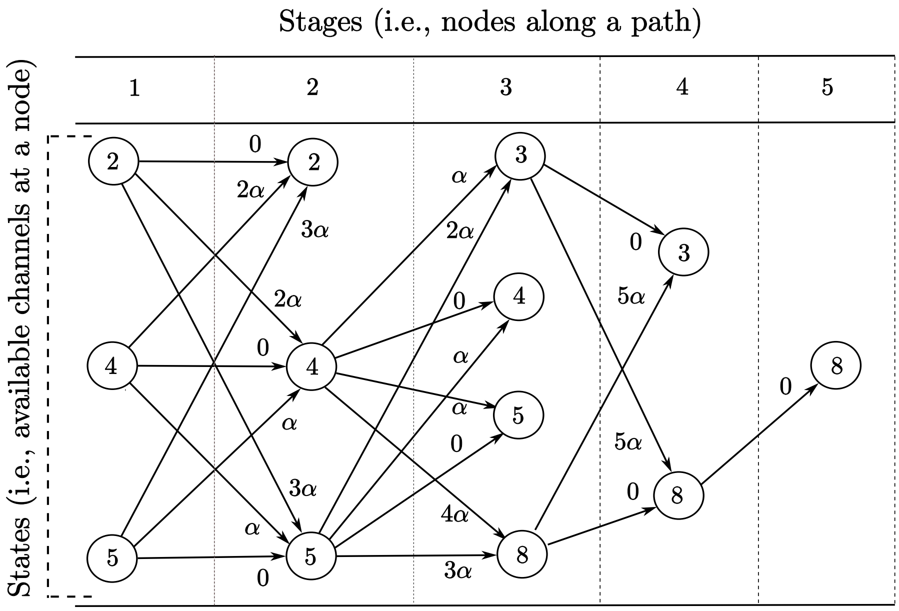


Figure 5.3: The dynamic program formulation of the path in Figure 5.2.

Choosing state m at stage i ($i < |\mathcal{R}|$) means that MR i receives data from MR $i+1$ on channel m. Finally, the dynamic program is outlined in Table 5.2. Equation (5.6) finds the optimal channel assignment, i.e., the one with the minimum end-to-end delay, along the entire route. Equation (5.7), on the other hand, finds the optimal channel assignment starting from MR i and ending at MR 1, such that MR i receives from MR $i+1$ on channel m. A boundary condition that gives all channel assignments equal cost of "0" at MR 1 is defined in (5.8). Figure 5.3 shows the formulation of the path given in Figure 5.2 excluding the packet transmission time (because it was assumed to be the same on all links and it, therefore, does not affect the selected channels).

5.4.2 Distributed optimal allocation

We devise a distributed implementation based on the above dynamic program formulation, that can be used in cognitive radio networks. To setup a connection from the source MR $|\mathcal{R}|$ to a destination MR 1 along the path $\mathcal{R}$, two phases are executed; the *forward phase* and the *backward phase* as summarized in Figure 5.4. In the forward phase, which starts from the destination and ends at the source, the dynamic program proposed earlier is used to calculate the optimal cost (end-to-end delay) along the route. In the backward phase, which starts from the source and ends at the destination, the channels to be used at each link to obtain the calculated end-to-end delay are identified at each MR along the route. The assumption that the destination starts the channel allocation process is suitable for multicast routing. In the case of multicast routing, the source MR (which is the gateway) sends a JOIN_REQ packet to the destination MR (the MR that should join a multicast session) on any available route (like an existing control route). Then, the destination MR starts a search process to find the optimal (minimum end-to-end delay) path to the gateway. That is why we assume the forward phase is initiated by the destination MR (for more details see Section 5.5). The forward phase operates as follows,

- The destination calculates the value $f(1, m)\ \forall m \in \mathcal{L}_1$ (see Table 5.2) and sends it in one packet to the next hop along the path, i.e., MR 2, through the CCC. We call this packet the *allocation cost packet (ACP)*. Let $\mathbf{ACP^{(i)}}$ denote the ACP packet sent by MR i to MR $i+1$, and $\mathbf{ACP^{(i)}}\cdot cost(m)$ is a field of $\mathbf{ACP}^{(i)}$ used to carry the cost of using channel m for the link $(i \rightarrow i+1)$. The source MR, i.e., MR 1, fills the fields of its ACP packet as follows:

$$\mathbf{ACP^{(1)}}\cdot cost(m) = 0, \qquad \forall m \in \mathcal{L}_1 \quad (5.1)$$

- Then, upon receiving $\mathbf{ACP^{(i-1)}}$ $(i < |\mathcal{R}|)$, MR i prepares its ACP packet as follows:

$$\mathbf{ACP^{(i)}}\cdot cost(m) = \min_{n \in \mathcal{L}_i \cap \mathcal{L}_{i-1}} C_{n,m}^{i-1,i} + \mathbf{ACP^{(i-1)}}\cdot cost(n) \quad (5.2)$$

- Finally, when the source MR, i.e, MR $|\mathcal{R}|$, receives the ACP packet of MR $|\mathcal{R}|-1$, it calculates $f^*(|\mathcal{R}|)$ as follows:

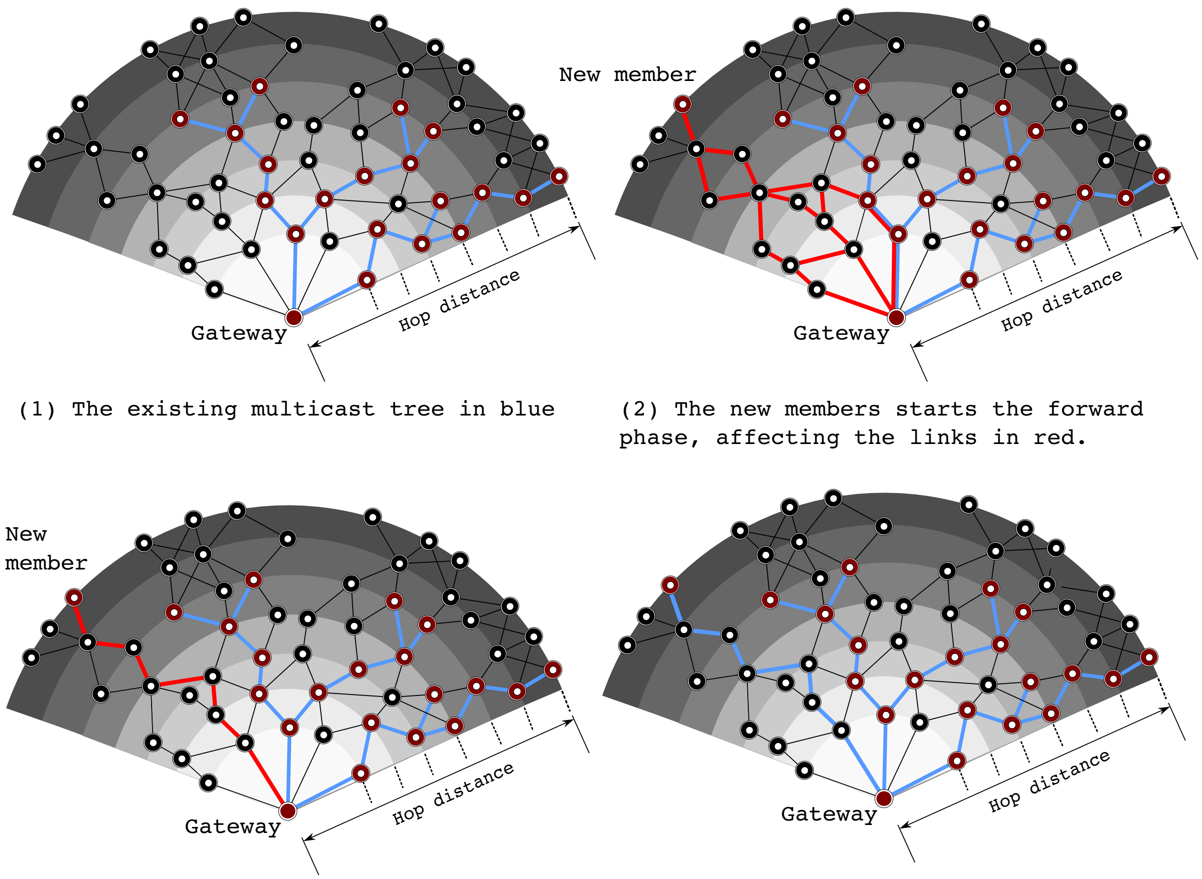
$$f^*(|\mathcal{R}|) = \min_{n \in \mathcal{L}_{|\mathcal{R}|} \cap \mathcal{L}_{|\mathcal{R}|-1}} L_n^{|\mathcal{R}|,|\mathcal{R}|-1} + \mathbf{ACP^{(|\mathcal{R}|-1)}}\cdot cost(n) \quad (5.3)$$

The backward phase, on the other hand, operates as follows,

- The source MR identifies the optimal channel on the link between $|\mathcal{R}|$ and $|\mathcal{R}|-1$. Let this channel be $k^*_{|\mathcal{R}|,|\mathcal{R}-1|}$, then

$$k^*_{|\mathcal{R}|,|\mathcal{R}|-1} = \operatorname*{argmin}_{n \in \mathcal{L}_{|\mathcal{R}|} \cap \mathcal{L}_{|\mathcal{R}|-1}} L_n^{|\mathcal{R}|,|\mathcal{R}|-1} + \mathbf{ACP^{(|\mathcal{R}|-1)}}\cdot cost(n) \quad (5.4)$$

  The source node $|\mathcal{R}|$ informs the previous node on the route, i.e, $|\mathcal{R}-1|$, about the identified channel.



(1) The existing multicast tree in blue

(2) The new members starts the forward phase, affecting the links in red.

(3) The backward phase initiated by the gateway, affecting the links in red.

(4) The new multicast tree after admitting the new member.

Figure 5.4: An illustrative example of the forward and backward phases of the route establishment process.

- Then, at each MR i on the route, for $1 < i < |\mathcal{R}|$, the channel $k^*_{i,i-1}$ is identified as follows,

$$k^*_{i-1,i} = \underset{n \in \mathcal{L}_i \cap \mathcal{L}_{i-1}}{\text{argmin}} \left(C^{i-1,i}_{n,k^*_{i,i+1}} + \mathbf{ACP^{(i-1)}} \cdot cost(n) \right) \tag{5.5}$$

- Finally, the destination, i.e., MR 1, receives, from MR 2, information about the channel assigned to the link $(2 \rightarrow 1)$.

The distributed implementation described above requires all MRs along $\mathcal{R}$ to maintain the ACP packets they receive until they hear back from MR $|\mathcal{R}|$. This also requires defining a timeout period such that if, within which, an MR does not hear back from MR $|\mathcal{R}|$, the ACP packet will be discarded.

Table 5.2: A Dynamic program for optimal channel allocation along a route $\mathcal{R}$

$$f^*(|\mathcal{R}|) = \min_{m \in \mathcal{L}_{|\mathcal{R}|} \cap \mathcal{L}_{|\mathcal{R}|-1}} L_m^{|\mathcal{R}|,|\mathcal{R}|-1} + f^*(|\mathcal{R}|-1, m) \tag{5.6}$$

$$f^*(i,m) = \min_{n \in \mathcal{L}_i \cap \mathcal{L}_{i-1}} \left(C_{n,m}^{i-1,i} + f^*(i-1,n) \right) \tag{5.7}$$

$$f^*(1,m) = 0, \ \forall m \in \mathcal{L}_1 \tag{5.8}$$

5.5 Multicast Routing: Challenges and Solutions

In this section, we use the dynamic program developed in Section 5.4 for the case of a single path as a building block to design an on-demand multicast routing and channel allocation algorithm. A common multicast routing design in wireless mesh networks is the tree-based structure, in which a multicast tree originates from the source of the multicast session and reaches every member of that session. We adopt this structure in this work. We introduce a decentralized dynamic tree construction algorithm by which an MR may attach itself to an existing multicast tree (or be the first in a new one) while jointly minimizing the end-to-end delay and throughput wastage at MRs along the selected route.

To understand the complexity of the problem, consider an MR, say i, that wants to join a multicast session. For i to find the route with the minimum end-to-end delay that connects it to the existing multicast tree (which could be only the gateway at the beginning), it must inspect all possible routes. This inspection is more complicated in cognitive radio networks than it is in traditional wireless networks because of the fact that *"longer paths, in terms of number of hops, do not necessarily impose longer delays because of the channel switching delay"*. In such a case, the search domain may include the entire network. Therefore, we need a systematic way to constrict the search domain and achieve a near optimal solution. A widely used technique is what we call the *level-based* approach which is based on the shortest

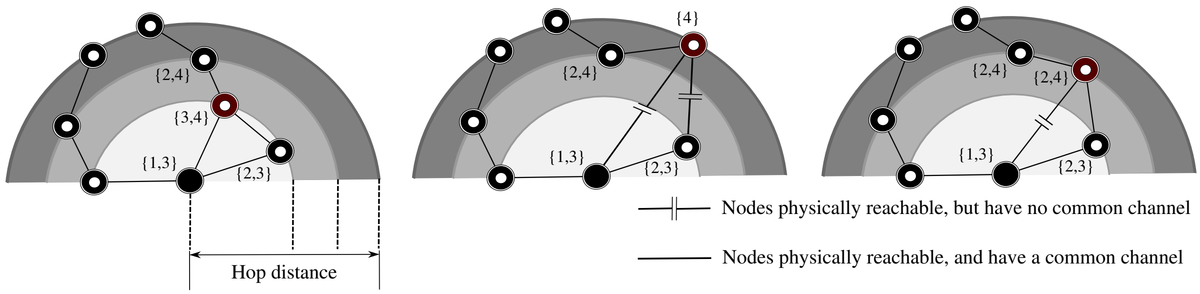


Figure 5.5: An example to illustrate how channel availability can affect the hop distance between SUs.

hop-count to the gateway. Based on this approach, each MR maintains its hop-count (on the shortest route) to the gateway. This can be achieved using a breadth first search (BFS) that starts from the gateway and traverses the whole network. It is worth pointing out that the hop-count distance should be updated regularly due to the varying channel availability. Consider the example in Figure 5.5

5.5.1 Finding the minimum hop distance (level) of MRs

To find the shortest hop-count distance,i.e., *level*, from the gateway to every other MR in the network, the gateway starts a search process on the CCC. Each MR i maintains a local level counter l_i, which is used to maintain the shortest hop distance to the gateway. This counter is initialized to ∞, except for the gateway for which the level is initialized to 0. Before describing the process of evaluating (and reconfiguring) the levels, we need to present some definitions. For an MR j in level l_j, define the following:

- *Neighbors of* j: the set of MRs that are within the transmission range of MR j, even if they do not share a common data channel with j. This set is denoted by $\mathcal{N}(j)$.

- *Parents of* j: the set of MRs in level $l_j{-}1$ that MR j can reach, which is denoted by $\mathcal{P}(j)$.

- *Children of* j: the set of MRs in level $l_j{+}1$ which can reach MR j , and is denoted by $\mathcal{C}(j)$.

It is worth pointing out that reachability between a pair of MRs requires two things. First, the two MRs are within the communication range of each other. Second, they share a common data channel (other than the CCC). The search process starts by having the gateway broadcasts a control packet, that we call LEVEL_UPDATE packet, with a *"level"* field initialized to 0, and a *"sender"* field initialized to 0 (the ID of the gateway MR) over the CCC.

Every MR, other than the gateway, runs the procedure shown in Algorithm 7. The algorithm guarantees that the level counters remain up-to-date for all MRs. The first *if*-statement (line 3) updates the level counter l_i (using the equation in line 6) upon receiving a LEVEL_UPDATE message from some MR $j \in \mathcal{N}(i)$. If the value of l_i is different from the current one, using the aforementioned equation, then all MRs in $\mathcal{N}(i)$ are notified about the new value (using the $SendLevelUpdate(\cdot)$ function as shown in line 7), and l_i is updated (line 8). If the channel availability at MR i, represented by the set of available channels $\mathcal{L}_i$, changes, then the second *if*-statement (line 9) will be executed. In that statement, MR i will first provide its neighbors, i.e., $\mathcal{N}(i)$, with the updated $\mathcal{L}_i$ (line 10). Then, it updates its level counter l_i and informs its neighbors about any change in l_i as shown in lines 12 and 13. Finally, the third *if*-statement (line 14) updates level l_i upon receiving an updated channel availability from some MR $j \in \mathcal{N}(i)$, and inform nodes in $\mathcal{N}(i)$ if l_i changes.

5.5.2 Multicast Routing Algorithm

The gateway sends a JOIN_REQ packet over the CCC to an MR(s) that should join a multicast group. This packet should contain enough information to identify the multicast session (like a group number). This packet is sent over the CCC. Upon receiving this packet, the MR needs to find the path that connects it with the multicast tree of the intended session with the minimum end-to-end delay. The search for this shortest path involves allocating channels to the links of that path for which no channels have been allocated yet (by other flows). At the same time, the bandwidth wastage due to channel switching should be kept as low as possible.

Before proposing the algorithm, we need to define a cost metric that can jointly represent the delay (transmission and switching) and bandwidth wastage. For the single path case in Section 5.4, the cost metric $C_{n,m}^{i-1,i}$ included the switching delay and transmission time only, because we did not address existing flows that pass through an MR. Let $\overline{\mathcal{L}}(i) \subseteq \mathcal{L}_i$ be the set

Algorithm 7: Level Evaluation and Reconfiguration

1 For an MR i (other than the gateway):
2 **begin**
3   **if** *a LEVEL_UPDATE message is received* **then**
4   j = LEVEL_UPDATE.*sender*;
5   l_j = LEVEL_UPDATE.*level*;
6         $$l_i^{new} = \begin{cases} l_j + 1 & \text{if } \mathcal{L}_i \cap \mathcal{L}_j \neq \emptyset \text{ and } l_i > l_j + 1 \\ l_i & \text{otherwise} \end{cases}$$
7   `SendLevelUpdate(`i,l_i,l_i^{new}`)`;
8         $l_i = l_i^{new}$;
9   **if** $\mathcal{L}_i$ *changes* **then**
10   Provide all MRs in $\mathcal{N}(i)$ with the new set $\mathcal{L}_i$;
11         $$l_i^{new} = \min_{j \in \mathcal{N}(i) | \mathcal{L}_i \cap \mathcal{L}_j \neq \emptyset} \begin{cases} \infty & \text{if } \mathcal{P}(j) = \{i\} \\ l_j + 1 & \text{otherwise} \end{cases}$$
12   `SendLevelUpdate(`i,l_i,l_i^{new}`)` ;
13         $l_i = l_i^{new}$;
14   **if** *an updated* $\mathcal{L}_j$ *for some* $j \in \mathcal{N}(i)$ *is received* **then**
15   Find l_i^{new} as described in line 12;
16   `SendLevelUpdate(`i,l_i,l_i^{new}`)`;
17         $l_i = l_i^{new}$;
18 **end**
19 `SendLevelUpdate(`i,l_i^{old},l_i^{new}`)`
20   **if** $l_i^{new} \neq l_i^{old}$ **then**
21   Create a new LEVEL_UPDATE message;
22   Set LEVEL_UPDATE.*sender*=i;
23   Set LEVEL_UPDATE.*level*=l_i^{new};
24   Broadcast LEVEL_UPDATE over the CCC;
25 **end**

of channels used to handle (i.e., receive or transmit) the flows (unicast or multicast) that are served by MR i. Then, we define the function $\Delta_i(\cdot)$ that takes as an argument a set of channels $\mathbf{S}$, and returns the maximum possible switching delay between any pair of channels in $\mathbf{S} \cup \overline{\mathcal{L}}_i$. Therefore,

$$\Delta_i(\mathbf{S}) = d_{sw}(\max_{m \in \mathbf{S} \cup \overline{\mathcal{L}}_i} m, \min_{n \in \mathbf{S} \cup \overline{\mathcal{L}}_i} n) \tag{5.9}$$

Equation (5.9) is based on the assumption that the cognitive radio remains tuned to the last used channel. If a different radio management policy is used (like having the cognitive radio always tuned to a particular channel during idle times), a different definition of the $\Delta_i(\cdot)$ function is needed. By including the $\Delta_i(\cdot)$ function into the cost metric $C_{n,m}^{j,i}$, channels closer to the ones already allocated to some incoming or outgoing links of MR i will be preferable over others that are farther away in the spectrum. Therefore, the nodal delay due to switching between flows served by the MR on different channels will be less, which intuitively means less bandwidth wastage.

The proposed on-demand multicast routing and channel allocation (OMRA) algorithm is outlined in Algorithm 8, and is based on the forward and backward phases of the distributed implementation of the dynamic program proposed in Section 5.4.2. When the MR that is supposed to join a multicast session, say MR i, receives a JOIN_REQ packet from the gateway, it prepares for each MR $j \in \mathcal{P}(i)$ an ACP packet as described in Section 5.4.2 (equation (5.1)) and sends it to j over the CCC. Then, every other MR that receives an ACP packet, from one of its children, does the same. In other words, it prepares an ACP packet for each of its parents as described in Section 5.4.2 (equation (5.2)) and sends it over the CCC. Therefore, the gateway will receive multiple ACP packets that originated at MR i, each of which corresponds to a distinct path between i and the gateway. The gateway then chooses the path with the minimum cost and initiates the backward phase described in Section 5.4.2 along that path.

5.6 Other Routing Metrics

In this section, we give two examples to show the extensibility of the proposed routing algorithm in terms of using other routing metrics. Consider the case where it is required to find the best route in terms of a particular metric, other than the end-to-end delay, while at the same time enforcing a limit on the maximum tolerable switching latency at a node. Such a metric can be, for example:

1. *End-to-end throughput*: this metric aims at finding the route that has the maximum end-to-end throughput, i.e, the maximum minimum link throughput.

2. *Maximize the minimum link lifetime*: this metric aims at finding the route that has the maximum minimum link expected lifetime. The link expected lifetime is defined as the length of the period from the moment a channel becomes available on a particular link until it becomes unavailable due to PU activity (see Figure 5.6 for an example).

   This metric is important to avoid frequent rerouting. Longer lifetime for a particular

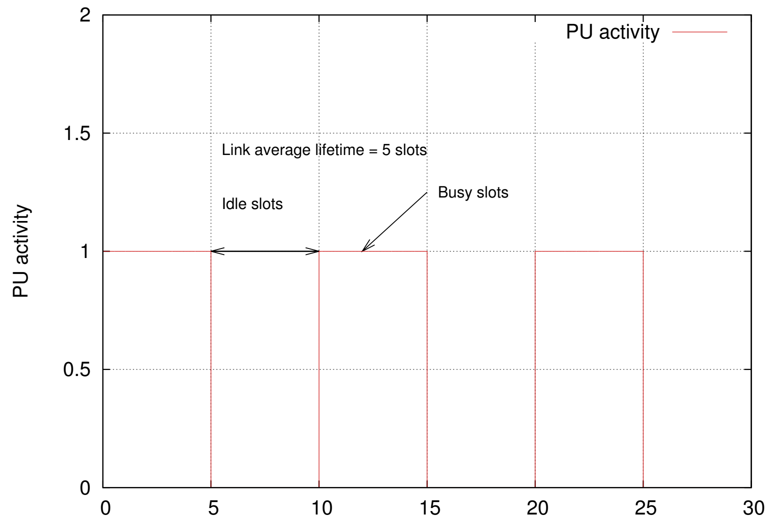


Figure 5.6: An example to explain the lifetime of a link on a give channel. The curve shows the PU usage of the channel across time slots (1 means the PU is using the channel).

link means longer periods between reroutings caused by failures of this link. Please note that the calculation of the link lifetime depends on how to define that a link has become unavailable. Unavailability for a single slot or a few slots, for example, may be tolerable and not sufficient to announce the end of the channel availability (i.e., lifetime) and trigger rerouting. Instead, a threshold, like say 10 slots, can be set as the cutoff point to announce channel unavailability. In any case, the calculation of the channel lifetime is independent from the routing decision as the former is just an input to the latter.

Let $r(i,j,m)$ denote the measure of a the routing metric on link (i,j,m), where i is the transmitter, j is the receiver, and m is the channel. Then, the following dynamic program can be used to find the optimal channel allocation along a route $\mathcal{R}$.

$$f^*(|\mathcal{R}|) = \max_{m \in \mathcal{L}_{|\mathcal{R}|} \cap \mathcal{L}_{|\mathcal{R}|-1}} \min\{r(|\mathcal{R}|, |\mathcal{R}|-1, m), f^*(|\mathcal{R}|-1, m)\} \tag{5.10}$$

$$f^*(i,m) = \max_{\substack{n \in \mathcal{L}_i \cap \mathcal{L}_{i-1} \\ d_{sw}(n,m) \le \zeta}} \min\{r(i, i-1, n), f^*(i-1, n)\} \tag{5.11}$$

$$f^*(1,m) = 0 \tag{5.12}$$

$d_{sw}(n,m)$ denotes the switching latency between channels n and m, and ζ denotes the maximum acceptable switching latency at a node. If we let $r(i,j,m)$ denote the link throughput, then this algorithm will find the channel allocation along the route that results in the maximum minimum link throughput. Similarly, if we let $r(i,j,m)$ denote the expected link lifetime, then this algorithm will find the channel allocation along the route which results in the maximum minimum link lifetime. Then, this algorithm can be extended to the multicast routing case in the same way we extended the end-to-end delay formulation.

Algorithm 8: On-Demand Multicast Routing and Channel Allocation (OMRA)

Input : Multicast group ID g, an MR i
Output: A multicast route with channel allocation.

if *i is the MR that has received the JOIN_REQ* **then**
  foreach $j \in \mathcal{P}(i)$ **do**
    Create a new ACP packet, $\mathbf{ACP^{(i)}}$;
    $\mathbf{ACP^{(i)}}\cdot cost(m) = \Delta(\{m\}),\ \forall m \in \mathcal{L}_i$;
    Send $\mathbf{ACP^{(i)}}$ to MR j;
else if *i is the gateway* **then**
  Find the optimal path from all received ACP's using (5.3);
  Start the backward phase (see Section 5.4.2);
else if $i \in \mathcal{T}(g)$ **then** /* On the multicast tree of g, i.e., $\mathcal{T}(g)$ */
  upon *receiving* $\mathbf{ACP^{(j)}}$ *from some* $j \in \mathcal{C}(i)$ **do**
    Create a new ACP packet, $\mathbf{ACP^{(i)}}$;
    Let k be the parent of i on the multicast tree;
    Let ν be the channel allocated to link (i,k);
    $\mathbf{ACP^{(i)}}\cdot cost(m) = \infty,\ \forall m \in \mathcal{L}_i/\{\nu\}$;
    $\mathbf{ACP^{(i)}}\cdot cost(\nu) = \min_{n \in \mathcal{L}_i \cap \mathcal{L}_j} \left(C^{j,i}_{n,\nu} + \mathbf{ACP^{(j)}}\cdot cost(n) \right)$
else /* Not on the multicast tree $\mathcal{T}(g)$ */
  upon *receiving* $\mathbf{ACP^{(j)}}$ *from some* $j \in \mathcal{C}(i)$ **do**
    Create a new ACP packet, $\mathbf{ACP^{(i)}}$;
    if *link (i,k) is allocated a channel ν* **then**
      $C^{j,i}_{n,m} = \infty, \forall m \in \mathcal{L}_i/\{\nu\}$;
      $C^{j,i}_{n,\nu} = d_{sw}(n,\nu) + L^{i,j}_n + \Delta_i(\{n,\nu\})$;
    else
      $C^{j,i}_{n,m} = d_{sw}(n,m) + L^{i,j}_n + \Delta_i(\{n,m\}), \forall m \in \mathcal{L}_i$;
    $\mathbf{ACP^{(i)}}\cdot cost(m) = \min_{n \in \mathcal{L}_i \cap \mathcal{L}_j} \left(C^{j,i}_{n,m} + \mathbf{ACP^{(j)}}\cdot cost(n) \right)$;
    Send $\mathbf{ACP^{(i)}}$ to all MRs $k \in \mathcal{P}(i)$;

5.7 Rerouting After Link Failures

Another important issue to address is the rerouting in case of link failures. In cognitive radio networks, link failures can, most of the time, mean that a PU is present on the channel and SUs have to vacate it. Depending on how frequently the PU reoccupies the channel, recalculating the route again by the disconnected members of the multicast tree can introduce significant overhead. We introduce a rerouting algorithm in this section that addresses the trade-off between route quality and the amount of overhead associated withe the route recovery process.

When a link on a route fails, the multicast member which created this route (i.e., initiated the forward phase) can start the route search process again by triggering the forward phase. Although this will result in the best possible solution under the OMRA algorithm, it will be a costly solution that produces lots of overhead especially with short channel lifetimes. Therefore, we need a solution that trades off the quality of the selected route for reducing the overhead of route reestablishment. The basic idea of our proposal is that if a node maintains the cost vector, i.e., the ACP packet, for each admitted route, it can then use it to resume the forward phase when the link between itself and its parent on a route breaks. Figure 5.7 shows an example of this process. There is a number of questions to answer regarding this recovery approach:

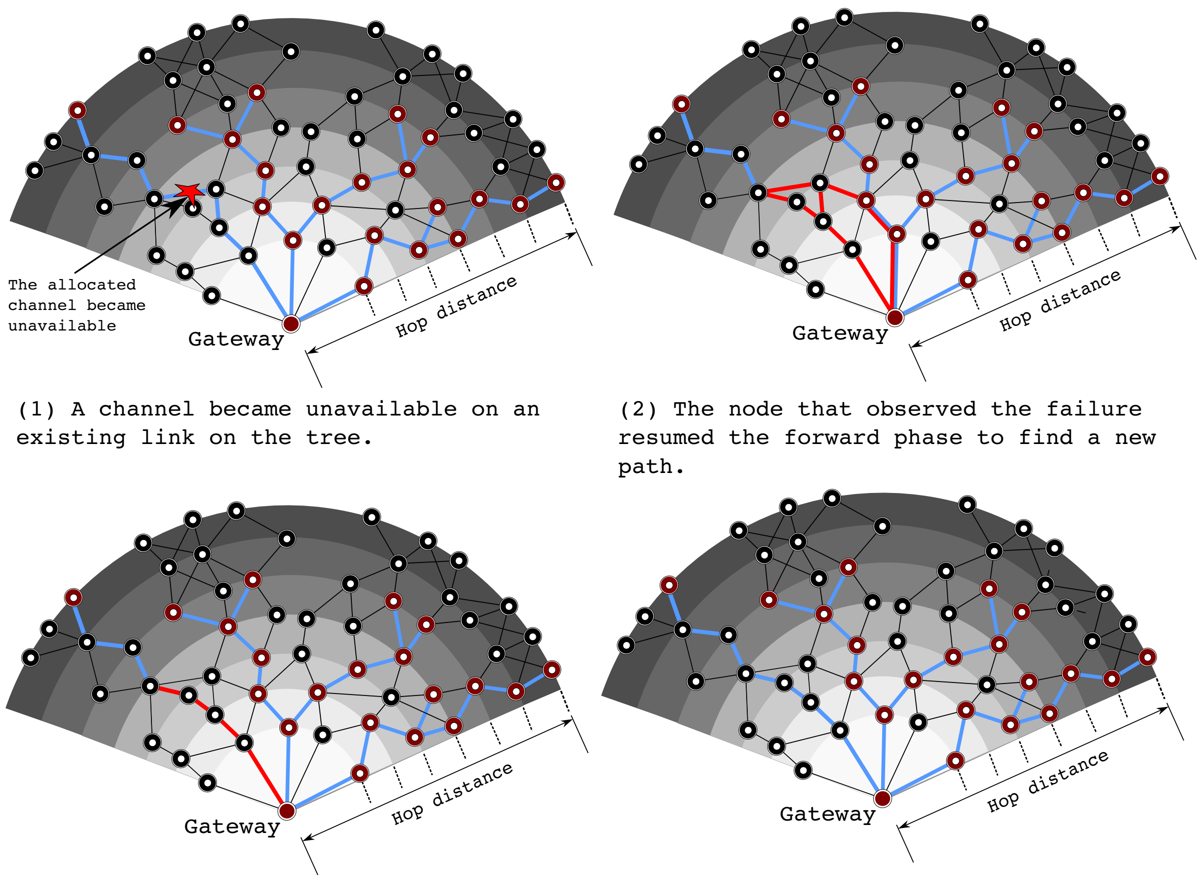
1. If a link between node i and its parent on channel k breaks, and this link serves multiple routes, how is node i going to recover all the routes together?

2. If more than one failure occur at the same time, how to avoid the redundant search done by the node closer to the gateway?

Let us first present the routing table that each node maintains. The table is presented in Figure 5.8, and it has the following columns:

1. *Originator*: the node that originated the route request (i.e., initiated the forward phase).

2. *Parent*: the parent of the current node on the route.



(1) A channel became unavailable on an existing link on the tree.

(2) The node that observed the failure resumed the forward phase to find a new path.

(3) The backward phase is initiated by the gateway, affecting the links in red.

(4) The failure recovery is complete.

Figure 5.7: An illustrative example of the failure recovery process.

3. *Child*: the child of the current node on the route.

4. *ACP*: the ACP cost packet that was received from the *child*.

The route recovery process is outlined in Algorithm 9. When a link $l = (n+1, n, k)$ that belongs to route $\mathcal{R}$ fails (where $n+1$ is the upstream node (closer to the MR), n is the downstream node, and k is the channel used on the link), node n runs the recovery algorithm. If n is the originator of $\mathcal{R}$ (i.e., $n = 0$) or $\mathcal{R}$ is the only route in n's routing table that is using link l, n resumes the forward phase as discussed earlier using the saved **ACP** packet [lines 4-7]. Otherwise, it informs the downstream node to recover the route [line 7]. This answers the first question we asked earlier, *"If a link between node i and its parent on channel k breaks,*

ORIGINATOR	PARENT	CHILD	ACP

Figure 5.8: The format of the routing table.

and this link serves multiple routes, how is node i going to recover all the routes together?" Unless the node is the originator of the route, it should never recover any route if multiple ones exist on the same link. Instead, it sends a notification downstream to recover the route. This downstream notification will continue until reaching a node that has only this single route served on the link with its upstream parent, or it is the originator of the route.

Lines 5 and 6 answer the second question "*If more than one failure occur at the same time, how to avoid the redundant search done by the node closer to the gateway?*" The node sends a notification downstream before starting the recovery process. If no downstream node initiates a recovery process within a timeout threshold, the current node assumes that there are no downstream failures and start the recovery. The timeout threshold is the round-trip latency between the current node and the originator of the route. This step is needed when the probability of multiple link failures on a route is high. If this is not the case, this step can be ignored and the current node can start the recovery process right away.

5.8 Performance Evaluation

5.8.1 OMRA routing

To evaluate the performance of the proposed routing algorithm, we study a cognitive radio mesh network of N MRs organized in a grid topology, and deployed in a square area of $A \times A$ meters. One of these N MRs is a gateway to the Internet, and is placed at the upper-left corner of the network field. Each MR can physically reach only the MRs in its left-hand, right-hand, upper, and lower cells in the grid (i.e., the communication range is $\frac{A}{\sqrt{N}}$). We have a total of K=10 licensed channels, each has an achievable rate of 10 Mbps. Channels are evenly spaced, and the separation between channels will be varied between 4 and 10 MHz. For all experiments in this subsection, N is set to 49, A to 500m, packet size is 1500 bytes, and the switching cost

Algorithm 9: The route recovery algorithm.

Input : Route $\mathcal{R}$
Broken link $l = (n+1, n, k)$ where n is the order of the downstream node on the route ($|\mathcal{R}|$ is the order of the gateway on the route).

Node n looks up the routing table record that corresponds to route $\mathcal{R}$, let that be $RT_n(\mathcal{R})$;
if $n = 0$ **then**
  n restarts the route search (forward phase) again.
else if $n > 0$ ***AND*** *route* $\mathcal{R}$ *is the only route in the routing table that is using link* l **then**
  n sends a notification downstream informing the nodes about the upcoming route recovery;
  if n *does not receive recovery updates for* $\mathcal{R}$ *within a timeout period (this depends on the length of the portion of the route from* n *to originator)* **then**
    n resumes the forward phase as described in section 5.5 using the saved ACP packet $RT_n(\mathcal{R}).ACP$;
else
  n informs $n-1$ on the route, i.e., $RT(\mathcal{R}).child$, to recover the path;

α to $1ms/10MHz$.

We compare the end-to-end delay between four different routing schemes: 1) OMRA routing where an MR investigate the possible paths through all of its parents (*OMRA-all*). 2) OMRA routing where an MR randomly picks one of its parent as the next hop on the path, and use the dynamic program to allocate channels along that path (*OMRA-one*). 3) Shortest path routing where an MR randomly picks one of its parents as the next hop on the path. The closest available channel (CAC) to that allocated to the link between the MR and its child, is allocated to the link between the MR and its parent. The member MR randomly selects the first channel from the its set of available channels. We denote this approach as *SPF-CAC*. 4) The last scheme is similar to the SPF-CAC except that the channel to be allocated to a link is chosen randomly. We denote this scheme as *SPF-RAND*.

We obtain average results for the end-to-end delay for the cases of single multicast session and multiple multicast sessions. Each point on the curves in Figures 5.9-5.14 corresponds to the average over 1500 randomly generated instances. In each instance, we vary the channel availability at each MR. An MR has channel k available with probability p and unavailable with probability $1 - p$. p is set to 0.393 for all experiments.

We first study the case of routing a single multicast session. Members of a session join the multicast group sequentially. The size of the multicast session is varied from 1 to 25. Figures 5.9, 5.10, and 5.11 show the average delay over all the members of a session for 4, 7, and 10 MHz channel spacing respectively. As the figures convey, the proposed OMRA algorithm, in its both variants, outperforms SPF-CAC and SPF-RAND. However, the gain is less when a single parent is explored at each MR (OMRA-one) than that when all parents are explored (OMRA-all). The SPF-CAC approach, on the other hand, is better that the SPF-RAND approach, and is close to the OMRA-one. However, they are all far outperformed by OMRA-all. Furthermore, as implied in Figures 5.9-5.11, the gain achieved by using the OMRA algorithm increases as the spacing between channels increases. For OMRA-all, the average gain over SPF-RAND is $\approx 23\%$ in the case of 4 MHz spacing, while it is $\approx 33\%$ in the case of 10 MHz spacing. Similarly, the gain of using OMRA-one is $\approx 10\%$ in the case of 4 MHz spacing, and $\approx 14\%$ in the case of 10 MHz spacing.

To evaluate the performance of the OMRA algorithm under the existence of multiple multicast sessions, we vary the number of sessions from 2 to 10, where each session has a size that is drawn uniformly at random from the range $[2, 15]$. Figures 5.12, 5.13, and 5.14 show the average end-to-end delay (over all sessions) for 4, 7, and 10 MHz spacings respectively. The results in these three figures confirm the superiority of the OMRA algorithm, in both of its variants, over the other approaches. Furthermore, the gain of using the OMRA algorithm (relative to the SPF-RAND approach) increases as the increase in the number of sessions in the network. For instance, the gain of OMRA-all increases from $\approx 25\%$ for 2 sessions to $\approx 29\%$ for 10 sessions at 4 MHz spacing, and from $\approx 35\%$ for 2 sessions to $\approx 40\%$ for 10 sessions at 10 MHz spacing.

Table 5.3: The mean times of the ON/OFF periods for all experiments

Physical hop distance	Mean time of the ON/OFF period
$[1-5]$	3min
$[6-14]$	5min
$[15-27]$	7min
$[28-44]$	9min

5.8.2 Route recovery

In this subsection, we evaluate the performance of the route recovery algorithm. We simulate the network in Figure 5.15. In this network, one gateway exists (in red), and one node (in blue) is trying to establish a route to the gateway using the OMRA-all algorithm (we refer to this node as the originator as it originates the route searching process). We randomize the channel availability at the nodes in the levels between the originator and the gateway to trigger rerouting. At each node in these levels, channel availability follows an ON-OFF model where the ON/OFF period follows an exponential distribution with means as summarized in Table 5.3. The availability of a particular channels is independent from other channels, and is independent across nodes as well. There are five channels, one of them is a CCC and the others are data channels each of which has a data rate of 10Mbps. The data packet size is 2272 bytes. Adjacent channels are separated by 10Mhz, while the channel switching factor is $1ms/10Mhz$.

We simulated the network for 24 hours, and compared two recovery approaches. The first is the one proposed in Section 5.7 where the recovery process starts from the point of failure. This one is referred to as "Reroute From Failure". The other approach is for the originator to start the recovery process, and this one is referred to as "Reroute From Originator".

Figure 5.16 shows the average number of ACP packets transmitted per second in the entire network during the forward phase using the settings in Table 5.3. As the figure shows, the overhead associated with the recovery from the point of failure is 45% less than that associated with the recovery from the originator. However, the average cost of the constructed route, as shown by Figure 5.17, is less when the recovery starts from the originator, as expected, but the difference is very small (around 5%). Figure 5.18 shows the actual cost of constructed routes over time.

Figure 5.19 shows the average number of ACP packets received by the gateway. This measures the average number of paths from the node that started the recovery process to the gateway. According to this Figure, the search domain when the recovery starts from the originator is around 36% wider than that when the recovery starts from the point of failure.

5.9 Chapter Summary

In this chapter, we studied the multicast routing and channel assignment problem in cognitive radio wireless mesh networks, with the objective of minimizing the end-to-end delay. A distributed on-demand routing and channel allocation algorithm that is based on dynamic programming was proposed. Numerical results show that the proposed algorithm outperforms other baseline routing and channel allocation schemes, namely, shortest path routing with "any-available-channel" allocation and shortest path with "closest-available-channel" allocation.

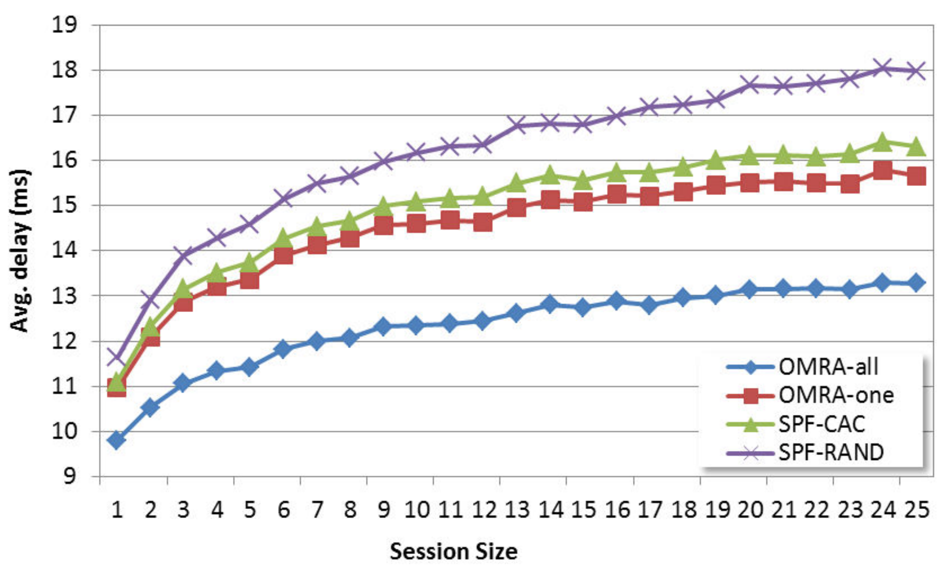


Figure 5.9: Average end-to-end delay for a single session with 4 MHz spacing.

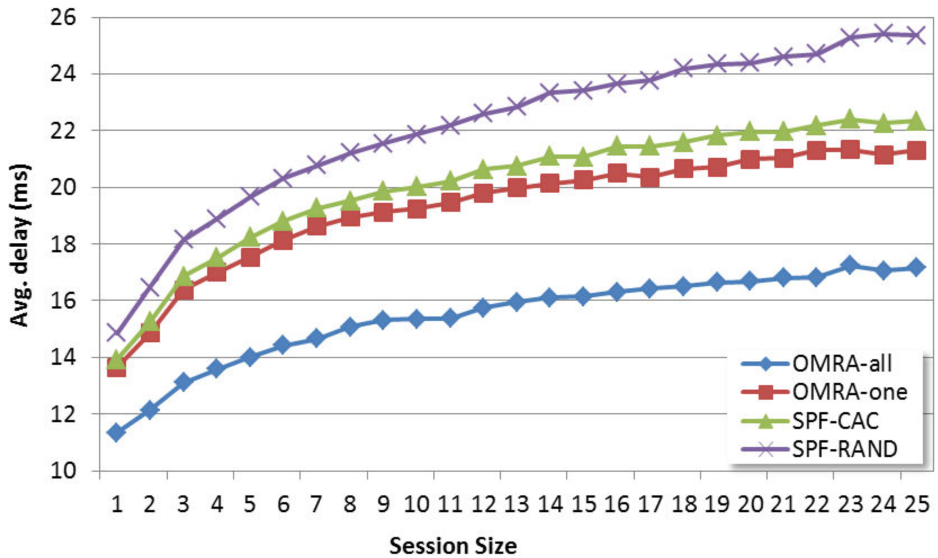


Figure 5.10: Average end-to-end delay for a single session with 7 MHz spacing.

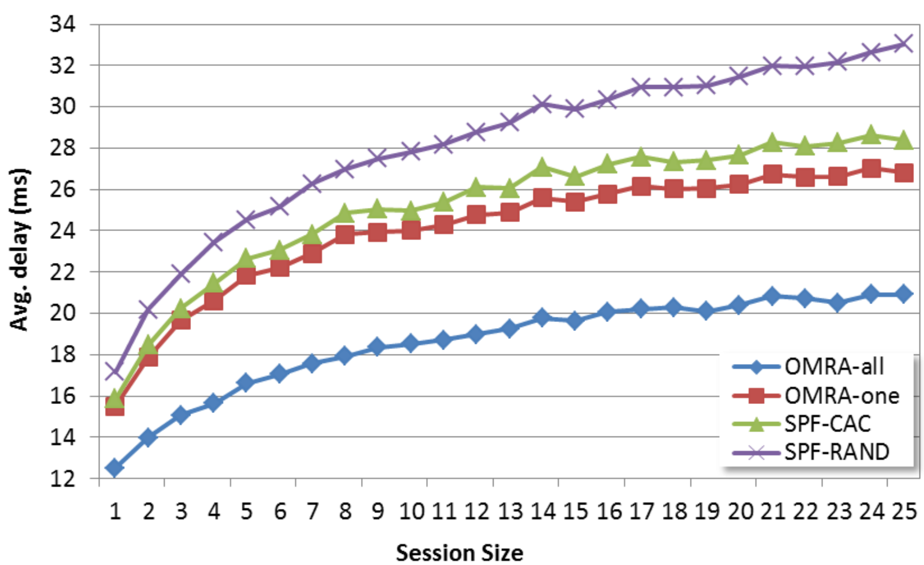


Figure 5.11: Average end-to-end delay for a single session with 10 MHz spacing.

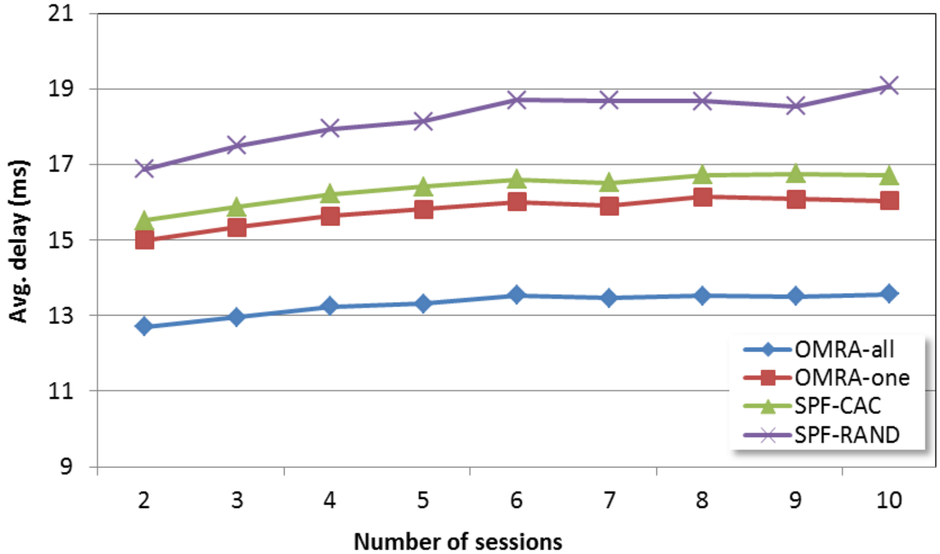


Figure 5.12: Average end-to-end delay for multiple sessions with 4 MHz spacing.

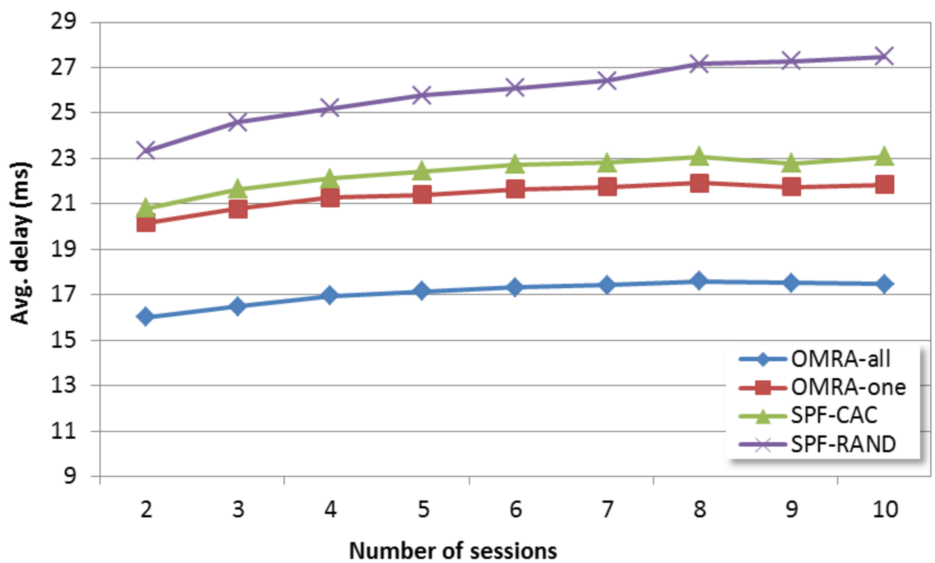


Figure 5.13: Average end-to-end delay for multiple sessions with 7 MHz spacing.

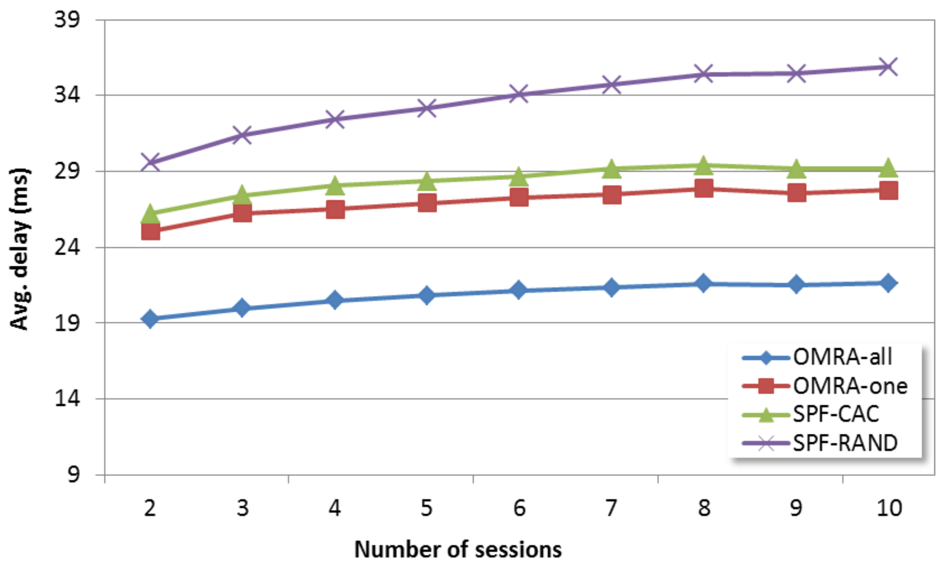


Figure 5.14: Average end-to-end delay for multiple sessions with 10 MHz spacing.

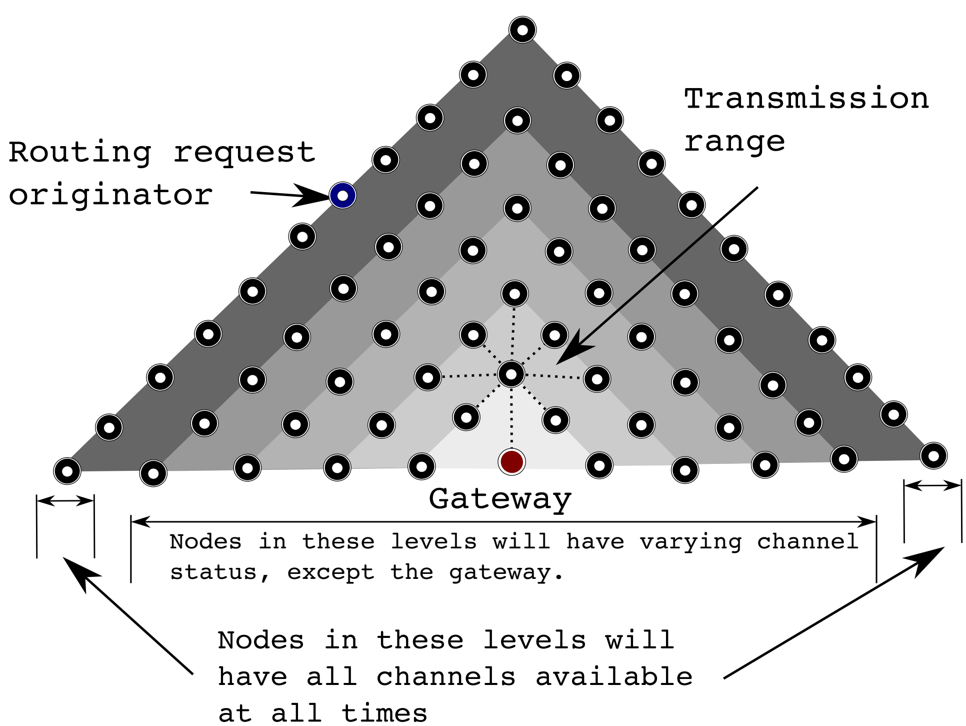


Figure 5.15: The network used to for the experiments in Section 5.8.2

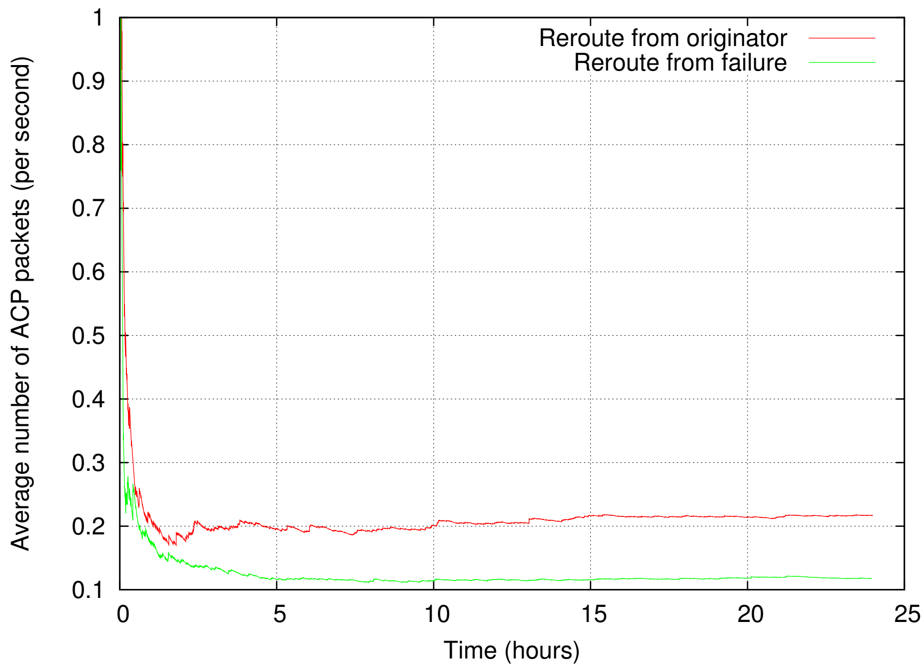


Figure 5.16: The average number of transmitted ACP packets during the forward phase in the entire network.

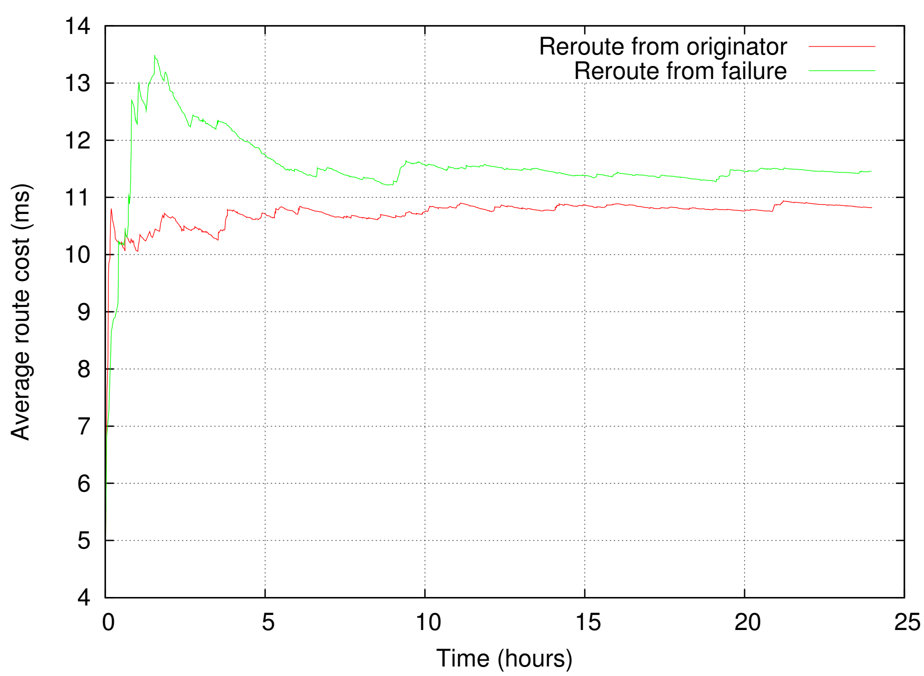


Figure 5.17: The average cost of the selected route reported at the gateway.

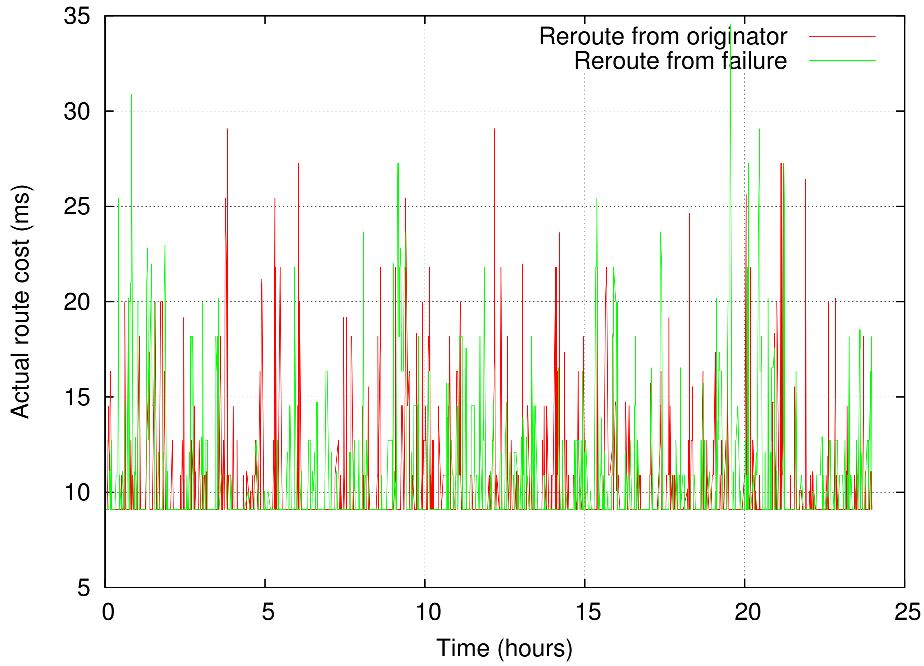


Figure 5.18: The actual cost of the selected route reported at the gateway.

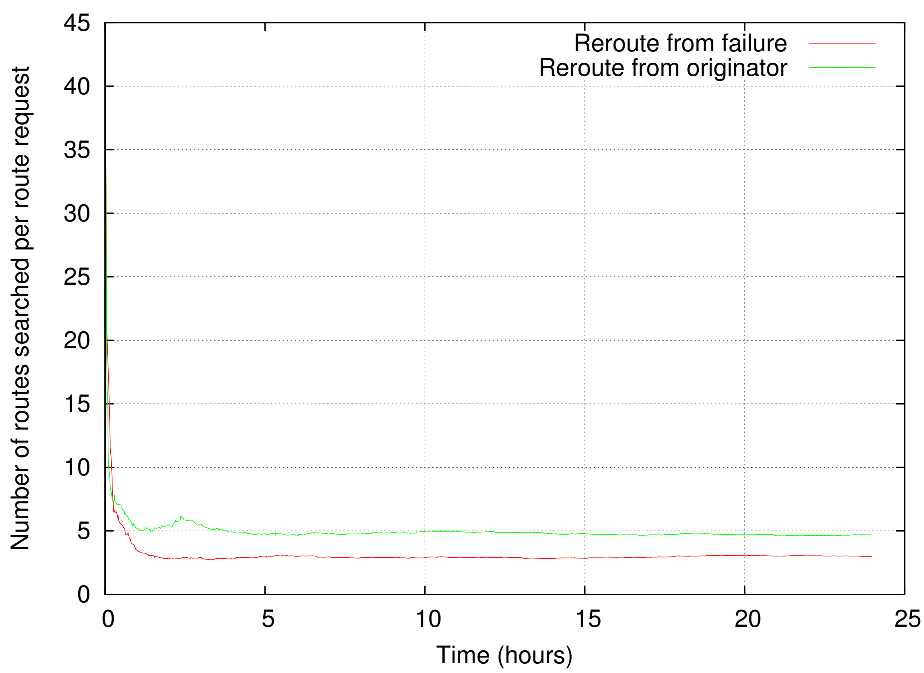


Figure 5.19: The average number of ACP packets received by the gateway for the same route.

CHAPTER 6. CONCLUSIONS AND FUTURE WORK

In this dissertation, we studied the channel allocation problem in cognitive radio wireless mesh networks. By controlling the tunability of the transmission and reception parts of the cognitive radio, four different modes of operation were defined for cognitive transceivers. Three channel allocation strategies based on the aforementioned modes were defined, namely receiver-based allocation *RBA*, transmitter-based allocation *TBA*, and all-tunable allocation *ATA*. MILP formulations were proposed RBA and ATA strategies with the objective of maximizing the number of served MCs with reliability guarantees on the uplink and downlink for each MC. Results show that the proposed RBA strategy outperforms the TBA and the ATA strategies even when a CCC is preassumed to exist. We also proposed a heuristic solution for the RBA problem. Results show that the accuracy of the proposed algorithm is, on average, within 28% of the optimal solution.

Furthermore, we studied the problem of assisted multicast scheduling in cognitive radio wireless mesh networks. We proposed an assistance paradigm that relies on receiver nodes to forward the multicast data to other receivers that have not yet received their own data. Furthermore, network coding was also proposed as another assistance technique that further reduced the total multicast time. Results show that the proposed assistance paradigm achieves a significant gain in reducing the total multicast period, i.e., enhancing the overall throughput.

Lastly, we studied the effect of channel switching on the multicast routing performance in wireless cognitive mesh networks. A generalized routing protocol based on dynamic programming was proposed. The proposed algorithm achieved a significant reduction in end-to-end delay compared to base-line approaches. We also proposed a route recovery algorithms that can reduce the amount of overhead associated with route reestablishment, and yet obtain good delay performance.

We plan to build on the observations and results from this dissertation to propose similar solutions for ad hoc cognitive radio networks. The ad hoc case is of course more challenging due to the lack of a central entity, an mesh route in the case of mesh networks, where decision to improve performance can be made.

APPENDIX A. The proof of theorem 4.4.1

Proof. A reduction from the set-cover problem can be easily drawn. The set cover problem has, as input, a universe $\mathcal{U}$ and a group of subsets $\mathcal{S} = \{S_1, \cdots, S_M\}$, and the objective is to find the minimum number of subsets that cover the universe $\mathcal{U}$, i.e., Minimize $|\mathcal{C}| : \mathcal{C} \subseteq \mathcal{S},\ \bigcup_{c \in \mathcal{C}} c = \mathcal{U}$. To map an instance of the set-cover problem into an instance of the UMS-single, we do the following:

- Create a hypothetical node n and mark it as the MR.
- For each member $u \in \mathcal{U}$ in the set-cover problem, create an MC u in the UMS-Single problem and extend an edge between u and n.
- Map each subset S_k in the set-cover problem into a channel k in UMS-Single problem. Then, make channel k available to every MC u *iff* $u \in S_k$.
- Make all channels available to the MR n.

Note that in the UMS-Single problem, MR n is the only transmitter and it transmits on one channel at each time slot. Also, note that any solution that has the MR transmits on the same channel in different time slots is not optimal, because the exact same set of MCs will receive the packet in both transmissions. Therefore, the minimum number of time slots to deliver the multicast packet to all MCs maps directly, by construction, to the minimum number of sets that can cover $\mathcal{U}$. In the other direction, the minimum number of subsets that cover the universe maps, also by construction, to the minimum number of time slots (because we use one channel per slot) needed to deliver the multicast packet.

□

APPENDIX B. The proof of Theorem 3.3.1

To prove the NP-hardness of the UDCP problem, we first prove the NP-completeness of a decision version of UDCP termed D-UDCP. The NP-completeness of the D-UDCP is proven by a reduction from the *Maximum Satisfiability* (MAX-SAT) problem. Before proceeding with the proof, we give the definitions of both the MAX-SAT and the D-UDCP problems.

Definition B.0.1 (MAX-SAT)**.** *Given a set of boolean variables* $\mathcal{X} = \{x_1, \cdots, x_N\}$*, each of which appears in at least one clause from the set of clauses* $\mathcal{C} = \{C_1, \cdots, C_M\}$*, where a clause is the OR operation of a number of variables each of which appears in either a negative of a positive form. For an integer* $k \leq M$*, is there a boolean assignment for the* N *variables that satisfies at least* k *clauses (i.e., make them evaluate to TRUE)? Let us denote this problem as* MAX-SAT$\langle \mathcal{X}, \mathcal{C}, k \rangle$*. The MAX-SAT problem is NP-complete [89].*

Definition B.0.2 (D-UDCP)**.** *Given the network of MRs as a graph* $G(\mathcal{B} \cup \mathcal{G}, E)$*, where* E *is the set of connectivity edges between MRs, and the channel availability at each MR (*$\mathcal{L}_i\ \forall i \in \mathcal{B} \cup \mathcal{G}$*). For an integer* $u \leq |\mathcal{B}|$*, is there a receiver-based channel assignment that guarantees both upstream and downstream connectivity for at least* u *non-gateway MRs? Let us denote this problem as* D-UDCP$\langle \mathcal{B}, \mathcal{G}, E, \mathcal{L}_i\ \forall i \in \mathcal{B} \cup \mathcal{G}, u \rangle$*.*

Lemma B.0.1. *D-UDCP* $\in$ **NP***.*

Proof. Let $\omega = |\mathcal{B}| + |\mathcal{G}| + 1$. Then, after adding a hypothetical node to the graph, the total number of directed edges, which result from the receiver-based channel assignment, in the graph is upper bounded by $\omega(\omega - 1)$. The following verifier is linear in the size of the graph (number of vertices plus number of edges).

Verifier D-UDCP$\langle \mathcal{B}, \mathcal{G}, E, \mathcal{L}_i\ \forall i \in \mathcal{B} \cup \mathcal{G}, u \rangle$:

1. Add a hypothetical node s to the graph and make it bi-connected with all nodes in $\mathcal{G}$. $O(1)$

2. Run a Depth-First-Search (DFS) algorithm starting from s and mark all reachable nodes as downstream connected. $O(\omega + \omega(\omega - 1) = O(\omega^2)$

3. Reverse all edges in the graph.

4. Run a Depth-First-Search (DFS) algorithm starting from s and mark all reachable nodes as upstream connected. $O(\omega + \omega(\omega - 1) = O(\omega^2)$

5. If the total number of nodes from $\mathcal{B}$ that are upstream AND downstream connected is greater than or equal to u then accept, otherwise reject. $O(|\mathcal{B}|)$

□

Lemma B.0.2. *MAX-SAT* $\leq_p$ *D-UDCP.*

Proof. Any instance of the MAX-SAT problem, say MAX-SAT$\langle \mathcal{X}, \mathcal{C}, k \rangle$, can be mapped into an instance of the D-UDCP problem as follows:

- Add a gateway MR g to $\mathcal{G}$ and define its set of available channels to be $\{0, 1\}$.

- Add a non-gateway MR for each variable, and a non-gateway MR for each clause to $\mathcal{B}$ with the set of available channels be $\{0, 1\}$ in both cases. Extend an edge between the gateway MR and every non-gateway MR that represents a variable, and add the edges to E.

- For each non-gateway MR that represents a clause, say C_i, add three non-gateway MRs n_i, b_i, and p_i which we call *auxiliary MRs* to $\mathcal{B}$. Then, extend edges between MR n_i (resp. p_i) and the MR that represents C_i as well as those which represent the variables that appear in a negative (resp. positive) form in the clause C_i and add them to E. Finally, extend edges between b_i and both n_i and p_i, and also add them to E. The sets of available channels for n_i, b_i, and p_i are $\{0\}$, $\{0, 1\}$, and $\{1\}$ respectively. We refer to set $\{n_i, b_i, p_i\}$ as the *auxiliary set* of the MRs that represents clause C_i.

- $u = 4 \times k + N$, where $N = |\mathcal{X}|$.

The above mapping can be written in a mathematical form as follows:

$\mathcal{G} = \{g\}$

$\mathcal{B} = \{x_i : x_i \in \mathcal{X}\} \cup \{C_i, n_i, b_i, p_i : C_i \in \mathcal{C}\}$

$E = \{(g, x_i) : x_i \in \mathcal{X}\} \cup \{(n_i, b_i), (b_i, p_i), (p_i, C_i), (n_i, C_i) : C_i \in \mathcal{C}\} \cup \{(n_i, x_j) : \overline{x}_j \in \mathcal{C}_i\} \cup \{(p_i, x_j) : x_j \in \mathcal{C}_i\}$

$\mathcal{L}_{n_i} = \{0\}$, $\mathcal{L}_{p_i} = \{1\}$, $\mathcal{L}_{b_i} = \mathcal{L}_{x_i} = \mathcal{L}_{C_i} = \{0, 1\}$

Note that all edges in E are now undirected. However, the solution will return directed edges due to the receiver-based nature of the channel assignment.

An example of this mapping process is shown in Figure B.1. Let an MR that represents a clause be denoted by CMR, an MR that represents a variable be denoted by VMR, and an auxiliary MR be denoted by AMR. According to the mapping procedure described above:

1. All VMRs are always upstream-and-downstream connected regardless of what channels are allocated to those VMRs and the gateway MR.

2. If at least one AMR from the auxiliary set of a CMR is upstream-and-downstream connected with the gateway, then there exists a channel allocation solution that guarantees all other AMRs of that set as well as the CMR to be upstream-and-downstream connected as shown in Figure B.2. It follows that if more than one AMR (i.e., two AMRs) are upstream-downstream connected, then the CMR and all AMRs in its auxiliary set are also upstream-and-downstream connected. It also follows that if neither the n_i AMR nor the p_i AMR is upstream-and-downstream connected, then the CMR C_i and all the AMRs in its auxiliary set are not upstream-and-downstream connected. In other words, a CMR and its auxiliary set are either upstream-and-downstream connected together, or not upstream-and-downstream together.

Suppose MAX-SAT$\langle \mathcal{X}, \mathcal{C}, k \rangle$ has a boolean assignment $\mathcal{X}^*$, where $\mathcal{X}^*(i) \in \{TRUE, FALSE\}$ is the boolean value of x_i, that satisfies the subset of clauses $\mathcal{C}^* \subseteq \mathcal{C}$ such that $|\mathcal{C}^*| \geq k$. Then, there must be a receiver-based channel assignment that guarantees both upstream and downstream connectivity for at least $u = 4 \times k + N$ non-gateway MRs. This can be easily proven as follows. Allocate either channel 0 or 1 to the gateway MR g. Then, for each variable x_i, if $\mathcal{X}^*(i)$=TRUE (resp. =FALSE) allocate channel 1 (resp. 0) to VMR x_i. This guarantees upstream and downstream connectivity to all the N VMRs. Then, allocate channel 0 to all n_i's and channel 1 to all p_i's. For each clause $C_i \in \mathcal{C}^*$, one of the following must hold true:

- At least one variable, say x_j, that appeared in a positive form in C_i has $\mathcal{X}^*(i)$ =TRUE. Therefore, VMR x_j, which is (by construction) a neighbor of p_i, must be allocated channel 1. As p_i is also allocated channel 1, and VMR x_j is upstream-and-downstream connected, then p_i is also upstream-and-downstream connected. As explained in Figure B.2, there exists a receiver-based channel allocation that makes CMR C_i and its auxiliary set all upstream-and-downstream connected.

- At least one variable, say x_j, that appeared in a negative form in C_i has $\mathcal{X}^*(i)$ =FALSE. Therefore, VMR x_j, which is (by construction) a neighbor of n_i, must be allocated channel 0. As n_i is also allocated channel 0, and VMR x_j is upstream-and-downstream connected, then n_i is also upstream-and-downstream connected. As explained in Figure B.2, there exists a receiver-based channel allocation that makes CMR C_i and its auxiliary set all upstream-and-downstream connected.

As the auxiliary set of each CMR is of size 3, the total number of upstream-and-downstream connected non-gateway MRs is $N + |\mathcal{C}^*| + 3 \times |\mathcal{C}^*| \geq N + 4 \times k$. Conversely, suppose D-UDCP$\langle \mathcal{B}, \mathcal{G}, E, u = 4 \times k + N \rangle$ has a receiver-based channel assignment $\mathcal{L}^*$, where $\mathcal{L}^*(i) \in \{0, 1\}$ is the channel allocated to MR i. Then, there is a boolean assignment to $\mathcal{X}$ such that at least k clauses from $\mathcal{C}$ are satisfied. Simply, for each VMR that is assigned channel 0 (*resp.* 1), assign the variable it represents a value of FALSE (*resp.* TRUE). Then, the same arguments made above can be used to show that at least k clauses will be satisfied. If a CMR C_i is upstream-and-downstream, then at least one of the two AMRs p_i and n_i must also be upstream-and-downstream connected. If p_i (*resp.* n_i) is upstream-and-downstream connected, then there must be at least one VMR that represents a variable, say x_j, that appeared in a positive (*resp.* negative) form in clause C_i which is assigned channel 1 (*resp.* channel 0). Given the channel-to-boolean mapping described earlier, clause C_i is satisfied in the MAX-SAT$\langle \mathcal{X}, \mathcal{C}, k \rangle$ instance. Therefore, the MAX-SAT$\langle \mathcal{X}, \mathcal{C}, k \rangle$ instance has at least $\frac{u-N}{4} \geq k$ satisfied clauses.

$\square$

This concludes the proof that D-UDCP is NP-complete. As UDCP is the optimization version of D-UDCP, then UDCP is NP-hard.

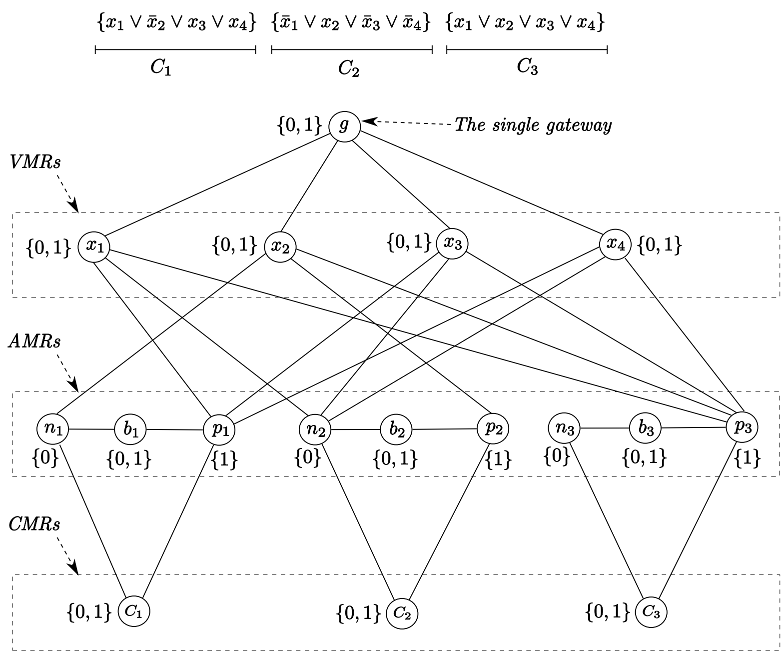


Figure B.1: Mapping a *maximum K-SAT* problem into an UDCP problem.

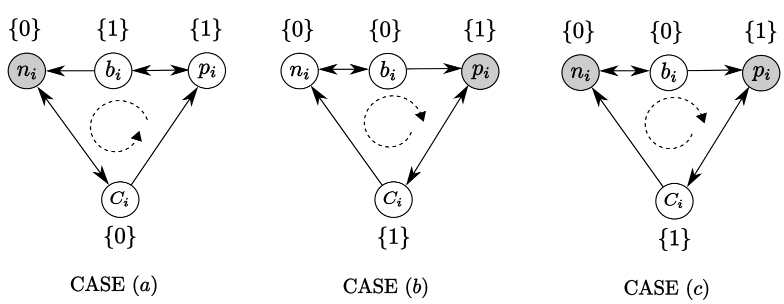


Figure B.2: A receiver-based channel allocation that can make the CMR i and its auxiliary set upstream-and-downstream connected given that either p_i, n_i, or both are upstream-and-downstream connected (gray colored).

APPENDIX C. MILP Formulations for TBA and ATA strategies

C.1 MILP formulation for the TBA problem

Let ν^k be a binary variable that is set to one if channel k is chosen as the CCC in the network. The channel allocation problem for the cognitive mesh network studied in this dissertation can be formulated as follows using the TBA strategy. $c_i^k = 1$ means, in this case, that channel k is assigned for node i for transmission (i.e., uplink).

$$Maximize \sum_{i \in \mathcal{A}} \sum_{k \in \mathcal{L}_i} c_i^k, \text{ subject to :}$$

(a) Channel assignment:

$\sum_{k \in \mathcal{L}} \nu^k \leq 1$ `/* One channel can be selected as the CCC. */`

$$\sum_{k \in \mathcal{L}_j} c_j^k \leq 1, \ \forall\ j \in \mathcal{B} \cup \mathcal{G} \cup \mathcal{A}$$ `/* At most one channel is allocated to each node (MC or MR). */`

$$\sum_{k \in \mathcal{L}_j} c_j^k \leq \sum_{k \in \mathcal{L}_i \cap \mathcal{L}_j} c_i^k, \ \forall\ i \in \mathcal{B} \cup \mathcal{G},\ j \in \mathcal{A}_i$$ `/* No channel is allocated to an MC unless its parent MR is allocated a channel that is shared between the two (to establish a downlink). */`

$$\sum_{k\in\mathcal{L}_j} c_j^k \leq \sum_{k\in\mathcal{L}_j} \nu^k,\ j\in\mathcal{A}$$

/* No channel is allocated to an MC unless a CCC is selected from its set of available channels. */

$$c_j^k + \nu^k \leq 1,\ \forall\, j\in\mathcal{A}\cup\mathcal{B}\cup\mathcal{G},\ k\in\mathcal{L}_j$$

/* The CCC cannot be allocated for data. */

(b) Upstream connectivity constraints:

$$\sum_{j:(i,j)\in E} f_{ij} - \sum_{j:(j,i)\in E} f_{ji} = 0,\ i\in\mathcal{B}\cup\mathcal{G}$$

/* Flow conservation constraint. */

$$\sum_{j:(s,j)\in E} f_{sj} = \sum_{j:(j,d)\in E} f_{jd} = \sum_{j\in\mathcal{A}}\sum_{k\in\mathcal{L}_j} c_j^k$$

/* Supply/demand constraint. */

$$f_{sj} = \sum_{i\in\mathcal{A}_j}\sum_{k\in\mathcal{L}_i} c_i^k,\ j\in\mathcal{B}\cup\mathcal{G}$$

/* Each MR receives an amount of flow equal to the number of MCs in its cell which have been allocated channels. */

$$\left.\begin{aligned} f_{ij} &\leq |\mathcal{A}|\cdot\sum_{k\in\mathcal{L}_i\cap\mathcal{L}_j} c_i^k,\ (i,j)\in E \\ f_{ij} &\leq |\mathcal{A}|\cdot\sum_{k\in\mathcal{L}_i\cap\mathcal{L}_j} \nu^k,\ (i,j)\in E \end{aligned}\right\}$$

/* Flow capacity constraints. For a flow to pass from MR i to MR j, the channel allocated to i must be available at j, and the selected CCC must be belong to the set of available channels of both i and j. */

(c) Downstream connectivity constraints:

$$\left.\begin{array}{l}
\sum_{j:(i,j)\in\overline{E}} g_{ij} - \sum_{j:(j,i)\in\overline{E}} g_{ji} = 0,\ i \in \mathcal{B}\cup\mathcal{G} \\
\sum_{j:(\overline{s},j)\in\overline{E}} g_{\overline{s}j} = \sum_{j:(j,\overline{d})\in\overline{E}} g_{j\overline{d}} = \sum_{j\in\mathcal{A}}\sum_{k\in\mathcal{L}_j} c_j^k \\
g_{j\overline{d}} = \sum_{i\in\mathcal{A}_j}\sum_{k\in\mathcal{L}_i} c_i^k,\ j\in\mathcal{B}\cup\mathcal{G} \\
g_{ij} \le |\mathcal{A}|\cdot \sum_{k\in\mathcal{L}_i\cap\mathcal{L}_j} c_i^k, (i,j)\in\overline{E} \\
g_{ij} \le |\mathcal{A}|\cdot \sum_{k\in\mathcal{L}_i\cap\mathcal{L}_j} v^k, (i,j)\in\overline{E}
\end{array}\right\}$$

`/* Similar to the constraints in the previous case, but now for the downstream flow commodity` g `instead of the upstream flow commodity` f`. */`

(d) Power control constraints:

$$\left.\begin{array}{l}
P_i^k \le P_r^{max}\cdot c_i^k,\ \forall\ i\in\mathcal{B}\cup\mathcal{G}, k\in\mathcal{L}_i \\
P_j^k \le P_c^{max}\cdot c_j^k,\ \forall\ j\in\mathcal{A},\ k\in\mathcal{L}_j
\end{array}\right\}$$

`/* A node cannot transmit on a channel unless it is allocated that channel. The transmission power must not exceed` P_r^{max} `(resp.` P_c^{max}`) for MRs (resp. MCs). */`

(e) Maximum inter-cell interference:

$$\zeta_{ij}^k \ge P_i^k \Psi_{ij}^k,\quad \forall i\in\mathcal{B}\cup\mathcal{G},\ j\in\mathcal{A}\cup\mathcal{B}\cup\mathcal{G}/(\mathcal{A}_i\cup\{i\}),\ k\in\mathcal{L}_i\cap\mathcal{L}_j.$$

$$\zeta_{ij}^k \ge P_m^k \Psi_{mj}^k,\quad \forall i\in\mathcal{B}\cup\mathcal{G},\ j\in\mathcal{A}\cup\mathcal{B}\cup\mathcal{G}/(\mathcal{A}_i\cup\{i\}),\ m\in\mathcal{A}_i,\ k\in\mathcal{L}_m\cap\mathcal{L}_j.$$

(f) Link reliability constraints:

Downlink:

$$\Psi_{ij}^{k} P_i^k - \gamma \left(N_0 + \sum_{m \in \mathcal{B} \cup \mathcal{G} - \{i\}} \zeta_{mj}^{k} \right) \geq \gamma (c_i^k + \sum_{w \in \mathcal{L}_j} c_j^w - 2) \left(N_0 + \sum_{m \in \mathcal{B} \cup \mathcal{G} - \{i\}} \zeta_{mj}^{max} \right),$$
$$\forall i \in \mathcal{B} \cup \mathcal{G}, j \in \mathcal{A}_i, k \in \mathcal{L}_i \cap \mathcal{L}_j$$

Uplink:

$$\Psi_{ji}^{k} P_j^k - \gamma \left(N_0 + \sum_{m \in \mathcal{B} \cup \mathcal{G} - \{i\}} \zeta_{mi}^{k} \right) \geq \gamma (c_j^k - 1) \left(N_0 + \sum_{m \in \mathcal{B} \cup \mathcal{G} - \{i\}} \zeta_{mi}^{max} \right),$$
$$\forall i \in \mathcal{B} \cup \mathcal{G}, j \in \mathcal{A}_i, k \in \mathcal{L}_i \cap \mathcal{L}_j$$

(f) Bounds:

$$c_j^k \in \{0,1\}, \quad j \in \mathcal{B} \cup \mathcal{G} \cup \mathcal{A};\ k \in \mathcal{L}_j$$

$$\nu^k \in \{0,1\}, \quad k \in \mathcal{L}$$

$$f_{ij} \geq 0, \quad (i,j) \in E;\ i,j \notin \{s,d\}$$

$$g_{ij} \geq 0, \quad (i,j) \in \bar{E};\ i,j \notin \{\bar{s},\bar{d}\}$$

$$g_{j\bar{d}},\ f_{sj} \geq 0, \quad j \in \mathcal{B}$$

$$g_{\bar{s}j}, f_{jd} \geq 0, \quad j \in \mathcal{G}$$

$$P_i^k \geq 0, \quad i \in \mathcal{B} \cup \mathcal{G} \cup \mathcal{A};\ k \in \mathcal{L}_i$$

The constraints in set (a) guarantees the following: at most one channel is selected as the CCC, at most one channel is assigned for each node to transmit on, no MC is assigned a channel unless its parent MR is assigned a channel, and that the channel used as a CCC is not assigned as a data channel to any node (MC or MR). The last two constraints in set (b) makes sure that no upstream flow is possible between MR i and MR j unless the CCC chosen is common between the two MRs and the channel assigned to i is shared with j. Similar constraints are used for the downstream connectivity case in set (c). Other constraints are similar to those used in the formulation of the RBA in Section 3.3.

C.2 MILP formulation for the ATA problem

Let c_{ji}^k be a binary variable that is set to 1 if channel k is assigned to the uplink (i, j) from MC i to its parent MR j. Also, let c_j^k be a binary variable that denotes the assignment of channel k to the downlink of MC j, and ν^k be a binary variable that is set to one if channel k is chosen as the CCC in the network. The ATA problem can be formulated as an MILP as follows.

$$\textit{Maximize} \sum_{i\in\mathcal{A}}\sum_{k\in\mathcal{L}_i} c_i^k, \quad \text{subject to :}$$

(a) Channel assignment:

$$\sum_{k\in\mathcal{L}} \nu^k \leq 1$$

/* One channel can be selected as the CCC. */

$$\sum_{k\in\mathcal{L}_j} c_j^k \leq \sum_{k\in\mathcal{L}_i\cap\mathcal{L}_j} c_{ji}^k, \ \forall\ i \in \mathcal{B}\cup\mathcal{G},\ j \in \mathcal{A}_i$$

/* No MC is assigned a channel on the downlink unless it has been assigned a channel on the uplink. */

$$\sum_{k\in\mathcal{L}_i\cap\mathcal{L}_j} c_{ji}^k \leq \sum_{k\in\mathcal{L}_i\cap\mathcal{L}_j} \nu^k, \ \forall\ i \in \mathcal{B}\cup\mathcal{G},\ j \in \mathcal{A}_i$$

/* No MC is assigned a channel on the uplink unless the selected CCC is common between that MC and its parent MR. */

$$\left.\begin{array}{ll} c_j^k + \nu^k \leq 1, & \forall\ j \in \mathcal{A},\ k \in \mathcal{L}_j \\ c_{ji}^k + \nu^k \leq 1, & \forall\ i \in \mathcal{B}\cup\mathcal{G},\ j \in \mathcal{A}_i,\ k \in \mathcal{L}_j\cap\mathcal{L}_i \end{array}\right\}$$

/* The channel selected as a CCC cannot be used for data. */

(b) Upstream connectivity constraints:

$$\sum_{j:(i,j)\in E} f_{ij} - \sum_{j:(j,i)\in E} f_{ji} = 0,\ i \in \mathcal{B} \cup \mathcal{G}$$ /* Flow conservation constraints. */

$$\sum_{j:(s,j)\in E} f_{sj} = \sum_{j:(j,d)\in E} f_{jd} = \sum_{j\in\mathcal{A}} \sum_{k\in\mathcal{L}_j} c_j^k$$ /* Supply/demand constraints. */

$$f_{sj} = \sum_{i\in\mathcal{A}_j} \sum_{k\in\mathcal{L}_i} c_i^k,\ j \in \mathcal{B} \cup \mathcal{G}$$ /* Each MR receives an amount of flow equal to the number of MCs in its cell which have been allocated channels. */

$$f_{ij} \le \begin{cases} |\mathcal{A}| \cdot \sum_{k\in\mathcal{L}_i\cap\mathcal{L}_j} \nu^k, & \text{if } |\mathcal{L}_i\cap\mathcal{L}_j|\ge 2,\ (i,j)\in E \\ 0, & \text{if } |\mathcal{L}_i\cap\mathcal{L}_j|\le 1,\ (i,j)\in E \end{cases}$$

/* For a flow to pass from MR i to MR j, the selected CCC must be shared between i and j as well as one other channel to use for data communication. */

(c) Downstream connectivity constraints:

$$\left.\begin{array}{l} \sum_{j:(i,j)\in \overline{E}} g_{ij} - \sum_{j:(j,i)\in \overline{E}} g_{ji} = 0,\ i \in \mathcal{B} \cup \mathcal{G} \\ \sum_{j:(\overline{s},j)\in \overline{E}} g_{\overline{s}j} = \sum_{j:(j,\overline{d})\in \overline{E}} g_{j\overline{d}} = \sum_{j\in\mathcal{A}} \sum_{k\in\mathcal{L}_j} c_j^k \\ g_{j\overline{d}} = \sum_{i\in\mathcal{A}_j} \sum_{k\in\mathcal{L}_i} c_i^k,\ j \in \mathcal{B} \cup \mathcal{G} \\ g_{ij} \le \begin{cases} |\mathcal{A}| \cdot \sum_{k\in\mathcal{L}_i\cap\mathcal{L}_j} \nu^k, & \text{if } |\mathcal{L}_i\cap\mathcal{L}_j|\ge 2,\ (i,j)\in \overline{E} \\ 0, & \text{if } |\mathcal{L}_i\cap\mathcal{L}_j|\le 1,\ (i,j)\in \overline{E} \end{cases} \end{array}\right\}$$

/* Similar to the constraints in the previous case, but now for the downstream flow commodity g instead of the upstream flow commodity f. */

(d) Power control constraints:

$$P_i^k \leq P_r^{max} \cdot \sum_{j:\{j \in \mathcal{A}_i,\ k \in \mathcal{L}_j\}} c_j^k,\ \forall\ i \in \mathcal{B} \cup \mathcal{G}, k \in \mathcal{L}_i$$

$$P_i^k \leq P_r^{max},\ \forall\ i \in \mathcal{B} \cup \mathcal{G}, k \in \mathcal{L}_i$$

$$P_j^k \leq P_c^{max} \cdot c_{ji}^k, \quad \forall\ i \in \mathcal{B} \cup \mathcal{G}, j \in \mathcal{A}_i,\ k \in \mathcal{L}_i \cap \mathcal{L}_j$$

(e) Maximum inter-cell interference:

$$\zeta_{ij}^k \geq P_i^k \Psi_{ij}^k, \forall i \in \mathcal{B} \cup \mathcal{G},\ j \in \mathcal{A} \cup \mathcal{B} \cup \mathcal{G}/(\mathcal{A}_i \cup \{i\}),\ k \in \mathcal{L}_i \cap \mathcal{L}_j$$

$$\zeta_{ij}^k \geq P_m^k \Psi_{mj}^k, \forall i \in \mathcal{B} \cup \mathcal{G},\ j \in \mathcal{A} \cup \mathcal{B} \cup \mathcal{G}/(\mathcal{A}_i \cup \{i\}),\ m \in \mathcal{A}_i,\ k \in \mathcal{L}_m \cap \mathcal{L}_j$$

(f) Link reliability constraints:

Downlink:
$\Psi_{ij}^k P_i^k - \gamma \left(N_0 + \sum_{m \in \mathcal{B} \cup \mathcal{G} - \{i\}} \zeta_{mj} \right) \geq \gamma (c_j^k - 1) \left(N_0 + \sum_{m \in \mathcal{B} \cup \mathcal{G} - \{i\}} \zeta_{mj}^{max} \right), \forall i \in \mathcal{B} \cup \mathcal{G}, j \in \mathcal{A}_i, k \in \mathcal{L}_i \cap \mathcal{L}_j$

Uplink:
$\Psi_{ji}^k P_j^k - \gamma \left(N_0 + \sum_{m \in \mathcal{B} \cup \mathcal{G} - \{i\}} \zeta_{mi} \right) \geq \gamma (c_{ji}^k + \sum_{w \in \mathcal{L}_j} c_j^w - 2) \left(N_0 + \sum_{m \in \mathcal{B} \cup \mathcal{G} - \{i\}} \zeta_{mi}^{max} \right), \forall i \in \mathcal{B} \cup \mathcal{G}, j \in \mathcal{A}_i, k \in \mathcal{L}_i \cap \mathcal{L}_j$

(f) Bounds:

$$c_j^k \in \{0,1\}, \quad i \in \mathcal{A},\ k \in \mathcal{L}_j$$

$$\nu^k \in \{0,1\}, \quad k \in \mathcal{L}$$

$$c_{ji}^k \in \{0,1\}, \quad i \in \mathcal{B} \cup \mathcal{G}, j \in \mathcal{A}_i,\ k \in \mathcal{L}_j \cap \mathcal{L}_i$$

$$\nu^k \in \{0,1\}, \quad k \in \mathcal{L}$$

$$f_{ij} \geq 0, \quad (i,j) \in E;\ i,j \notin \{s,d\}$$

$$g_{ij} \geq 0, \quad (i,j) \in \bar{E};\ i,j \notin \{\bar{s},\bar{d}\}$$

$g_{j\bar{d}},\ f_{sj} \geq 0, \quad j \in \mathcal{B}$

$g_{\bar{s}j}, f_{jd} \geq 0, \quad j \in \mathcal{G}$

$P_i^k \geq 0, \quad i \in \mathcal{B} \cup \mathcal{G} \cup \mathcal{A};\ k \in \mathcal{L}_i$

The constraints in set (a) guarantees the following: at most one channel is selected as the CCC, at most one channel is assigned for the downlink (c_j^k) and one for the uplink (c_{ji}^k), no downlink is assigned a channel unless the uplink is assigned one and a CCC that is shared between the MC and it parent MR is selected. The channel used as a CCC must not be assigned to any link (i.e., for data communication). The last constraint in set (b) guarantees that no upstream flow is possible between MR i and MR j unless the CCC chosen is shared between the two MRs and they have at least one more shared channel to use for data transmission. A similar constraint is used for the downstream connectivity case in set (c). Other constraints are similar to the those used in the formulation of the RBA problem in Section 3.3.

Bibliography

[1] Ian F. Akyildiz, Won-Yeol Lee, Mehmet C. Vuran, and Shantidev Mohanty. Next generation/dynamic spectrum access/cognitive radio wireless networks: a survey. *Computer Networks*, 50(13):2127–2159, 2006.

[2] III Mitola, J. and Jr. Maguire, G.Q. Cognitive radio: making software radios more personal. *IEEE Personal Communications*, 6(4):13–18, August 1999.

[3] Joseph Mitola. *An Integrated Agent Architecture for Software Defined Radio*. PhD thesis, Royal Institute of Technology (KTH), Stockholm, Sweden, May 2000.

[4] C. Raman, R.D. Yates, and N.B. Mandayam. Scheduling variable rate links via a spectrum server. In *First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, pages 110 –118, November 2005.

[5] O. Ileri, D. Samardzija, and N.B. Mandayam. Demand responsive pricing and competitive spectrum allocation via a spectrum server. In *First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, pages 194 –202, November 2005.

[6] S.A. Zekavat and X. Li. User-central wireless system: ultimate dynamic channel allocation. In *First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, pages 82 –87, November 2005.

[7] R.D. Yates, C. Raman, and N.B. Mandayam. Fair and efficient scheduling of variable rate links via a spectrum server. In *IEEE International Conference on Communications (ICC)*, volume 11, pages 5246 –5251, June 2006.

[8] J. Zhao, H. Zheng, and G.-H. Yang. Distributed coordination in dynamic spectrum allocation networks. In *First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, pages 259–268, November 2005.

[9] Lili Cao and Haitao Zheng. Distributed spectrum allocation via local bargaining. In *The Second Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks (SECON)*, pages 475 – 486, September 2005.

[10] Lili Cao and Haitao Zheng. On the efficiency and complexity of distributed spectrum allocation. In *The Second International Conference on Cognitive Radio Oriented Wireless Networks and Communications (CrownCom)*, pages 357 –366, August 2007.

[11] Yuan Wu and D.H.K. Tsang. Distributed power allocation algorithm for spectrum sharing cognitive radio networks with qos guarantee. In *IEEE INFOCOM*, pages 981 –989, April 2009.

[12] I.F. Akyildiz and Xudong Wang. A survey on wireless mesh networks. *IEEE Communications Magazine*, 43(9):S23–S30, September 2005.

[13] R. Hincapie, Li Zhang, Jian Tang, Guoliang Xue, R.S. Wolff, and R. Bustamante. Efficient recovery algorithms for wireless mesh networks with cognitive radios. In *IEEE International Conference on Communications (ICC)*, pages 1–5, June 2009.

[14] K.R. Chowdhury and I.F. Akyildiz. Cognitive wireless mesh networks with dynamic spectrum access. *IEEE Journal on Selected Areas in Communications*, 26(1):168–181, January 2008.

[15] Tao Chen, Honggang Zhang, G.M. Maggio, and I. Chlamtac. Cogmesh: A cluster-based cognitive radio network. In *Second IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, pages 168–178, April 2007.

[16] R.C. Pereira, R.D. Souza, and M.E. Pellenz. Using cognitive radio for improving the capacity of wireless mesh networks. In *IEEE Vehicular Technology Conference (VTC)*, pages 1–5, September 2008.

[17] Hisham M. Almasaeid and Ahmed E. Kamal. Receiver-based channel allocation for wireless cognitive radio mesh networks. In *IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, April 2010.

[18] Tao Chen, Honggang Zhang, M. Matinmikko, and M.D. Katz. Cogmesh: Cognitive wireless mesh networks. In *IEEE GLOBECOM Workshops*, pages 1–6, December 2008.

[19] Open spectrum: A path to ubiquitous connectivity. *Queue*, 1(3):60–68, 2003.

[20] D. Niyato and E. Hossain. Cognitive radio for next-generation wireless networks: an approach to opportunistic channel selection in ieee 802.11-based wireless mesh. *IEEE Wireless Communications*, 16(1):46–54, February 2009.

[21] P. Kyasanur and N. H. Vaidya. Capacity of multichannel wireless networks: Impact of channels, interfaces, and interface switching delay. Technical report, University of Illinois at Urbana-Champaign, Oct. 2006.

[22] P. Kyasanur and N. H. Vaidya. Routing in multi-channel multi-interface ad hoc wireless networks. Technical report, University of Illinois at Urbana-Champaign, 2004.

[23] Hisham M. Almasaeid and Ahmed E. Kamal. Assisted-multicast scheduling in wireless cognitive radio mesh networks. In *IEEE International Conference on Communications (ICC)*, May 2010.

[24] H.M. Almasaeid, T.H. Jawadwala, and A.E. Kamal. On-demand multicast routing in cognitive radio mesh networks. In *IEEE Global Telecommunications Conference (GLOBECOM)*, pages 1 –5, December 2010.

[25] F.F. Digham, M.-S. Alouini, and M.K. Simon. On the energy detection of unknown signals over fading channels. In *IEEE International Conference on Communications (ICC)*, volume 5, pages 3575–3579, May 2003.

[26] Junyang Shen, Yuanan Liu, Siyang Liu, Jinchun Gao, Gang Xie, and Caixia Chi. Robust energy detection based on bayesian estimation for cognitive radio. In *IEEE Global Telecommunications Conference (GLOBECOM)*, pages 1–5, December 2008.

[27] V. Turunen, M. Kosunen, A. Huttunen, S. Kallioinen, P. Ikonen, A. Parssinen, and J. Ryynanen. Implementation of cyclostationary feature detector for cognitive radios. In *Fourth International Conference on Cognitive Radio Oriented Wireless Networks and Communications (CROWNCOM)*, pages 1–4, June 2009.

[28] Jiang Zhu, Zhengguang Xu, Furong Wang, Benxiong Huang, and Bo Zhang. Double threshold energy detection of cooperative spectrum sensing in cognitive radio. In *Third International Conference on Cognitive Radio Oriented Wireless Networks and Communications (CrownCom)*, pages 1–5, May 2008.

[29] A. Fehske, J. Gaeddert, and J.H. Reed. A new approach to signal classification using spectral correlation and neural networks. In *First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, pages 144–150, November 2005.

[30] B. Ramkumar. Automatic modulation classification for cognitive radios using cyclic feature detection. *IEEE Circuits and Systems Magazine*, 9(2):27–45, 2009.

[31] Wei Zhang, R. Mallik, and K. Letaief. Optimization of cooperative spectrum sensing with energy detection in cognitive radio networks. *IEEE Transactions on Wireless Communications*, 8(12):5761 –5766, December 2009.

[32] A. Ghasemi and E.S. Sousa. Collaborative spectrum sensing for opportunistic access in fading environments. In *First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, pages 131–136, November 2005.

[33] A.R. Biswas, T.C. Aysal, S. Kandeepan, D. Kliazovich, and R. Piesiewicz. Cooperative shared spectrum sensing for dynamic cognitive radio networks. In *IEEE International Conference on Communications (ICC)*, pages 1–5, June 2009.

[34] Chunhua Sun, Wei Zhang, and K. Ben. Cluster-based cooperative spectrum sensing in cognitive radio systems. In *IEEE International Conference on Communications (ICC)*, pages 2511 –2515, June 2007.

[35] Junyang Shen, Tao Jiang, Siyang Liu, and Zhongshan Zhang. Maximum channel throughput via cooperative spectrum sensing in cognitive radio networks. *IEEE Transactions on Wireless Communications*, 8(10):5166 –5175, October 2009.

[36] J. Unnikrishnan and V.V. Veeravalli. Cooperative spectrum sensing and detection for cognitive radio. In *IEEE Global Telecommunications Conference (GLOBECOM)*, pages 2972–2976, November 2007.

[37] R. Thobaben and E.G. Larsson. Sensor-network-aided cognitive radio: On the optimal receiver for estimate-and-forward protocols applied to the relay channel. In *Conference Record of the Forty-First Asilomar Conference on Signals, Systems and Computers, (ACSSC)*, pages 777 –781, November 2007.

[38] V.Fodor, R.Thobaben, B.Mercier, et al. Sensor networks for cognitive radio: Theory and system design. In *ICT Mobile Summit*, June 2008.

[39] V. Fodor, I. Glaropoulos, and L. Pescosolido. Detecting low-power primary signals via distributed sensing to support opportunistic spectrum access. In *IEEE International Conference on Communications (ICC)*, pages 1–6, June 2009.

[40] SENDORA project. http://www.sendora.eu.

[41] Tao Shu and Marwan Krunz. Truthful least-priced-path routing in opportunistic spectrum access networks. In *IEEE INFOCOM*, March 2010.

[42] A. Alsarhan and A. Agarwal. Cluster-based spectrum management using cognitive radios in wireless mesh network. In *The 18th International Conference on Computer Communications and Networks (ICCCN)*, pages 1–6, August 2009.

[43] W. Kim, S. Oh, M. Gerla, and J.-S. Park. Cocast: Multicast mobile ad hoc networks using cognitive radio. *IEEE MILCOM*, 2009.

[44] T. Chen, H. Zhang, G.M. Maggio, and I. Chlamtac. Topology management in cogmesh: A cluster-based cognitive radio mesh network. In *IEEE International Conference on Communications (ICC)*, pages 6516–6521, June 2007.

[45] Y. R. Kondareddy and P. Agrawal. Synchronized mac protocol for multi-hop cognitive radio networks. In *IEEE International Conference on Communications (ICC)*, pages 3198–3202, May 2008.

[46] Y. R. Kondareddy and P. Agrawal. Selective broadcasting in multi-hop cognitive radio networks. *IEEE Sarnoff Symposium*, pages 1–5, April 2008.

[47] Wendong Hu, D. Willkomm, M. Abusubaih, J. Gross, G. Vlantis, M. Gerla, and A. Wolisz. Cognitive radios for dynamic spectrum access - dynamic frequency hopping communities for efficient ieee 802.22 operation. *IEEE Communications Magazine*, 45(5):80 –87, May 2007.

[48] Xiaohua Li and Juite Hwu. A frequency hopping spread spectrum transmission scheme for uncoordinated cognitive radios. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2345 –2348, April 2009.

[49] Donglin Hu, Shiwen Mao, and J.H. Reed. On video multicast in cognitive radio networks. In *IEEE INFOCOM*, pages 2222–2230, April 2009.

[50] Duy (Danny) Ngo and Chintha Tellambura. Resource allocation for ofdm-based cognitive radio multicast networks. In *IEEE Wireless Communications and Networking Conference (WCNC)*, Budapest, Hungary, April 5-8 2009.

[51] Cheng Wang, Shaojie Tang, Xiang-Yang Li, and Changjun Jiang. Multicast capacity of multihop cognitive networks. In *IEEE 6th International Conference on Mobile Adhoc and Sensor Systems (MASS)*, pages 274 –283, October 2009.

[52] R. Ahlswede, Ning Cai, S.-Y.R. Li, and R.W. Yeung. Network information flow. *IEEE Transactions on Information Theory*, 46(4):1204 –1216, July 2000.

[53] Christina Fragouli, Jean-Yves Le Boudec, and Jörg Widmer. Network coding: an instant primer. *SIGCOMM Comput. Commun. Rev.*, 36(1):63–68, 2006.

[54] D.S. Lun, N. Ratnakar, R. Koetter, M. Medard, E. Ahmed, and Hyunjoo Lee. Achieving minimum-cost multicast: a decentralized approach based on network coding. In *IEEE INFOCOM*, volume 3, pages 1607–1617, March 2005.

[55] T. Ho, M. Medard, R. Koetter, D.R. Karger, M. Effros, Jun Shi, and B. Leong. A random linear network coding approach to multicast. *IEEE Transactions on Information Theory*, 52(10):4413 –4430, October 2006.

[56] O.M. Al-Kofahi and A.E. Kamal. Network coding-based protection of many-to-one flow networks. In *IEEE Internatonal Conference on Mobile Ad hoc and Sensor Systems (MASS)*, pages 1–10, October 2007.

[57] O.M. Al-Kofahi and A.E. Kamal. Network coding-based protection of wireless mesh networks. In *International Symposium on High Capacity Optical Networks and Enabling Technologies (HONET)*, pages 1 –8, November 2007.

[58] O. M. Al-Kofahi and A. E. Kamal. Network coding-based protection of many-to-one wireless flows. *IEEE Journal of Selected Areas on Communications special issue on Network Coding in Wireless Communications*, 27(5):797–813, 2009.

[59] Shengli Zhang, Soung Chang Liew, and Patrick P. Lam. Hot topic: physical-layer network coding. In *Proceedings of the 12th annual international conference on Mobile computing and networking (MobiCom)*, pages 358–365, New York, NY, USA, 2006. ACM.

[60] Sachin Katti, Shyamnath Gollakota, and Dina Katabi. Embracing wireless interference: analog network coding. In *Proceedings of the 2007 conference on Applications, technologies, architectures, and protocols for computer communications (SIGCOMM)*, pages 397–408, New York, NY, USA, 2007. ACM.

[61] Lijuan Geng, Ying-Chang Liang, and F. Chin. Network coding for wireless ad hoc cognitive radio networks. *IEEE 18th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, pages 1–5, September 2007.

[62] C.-H. Huang, Y.-C. Lai, and K.-C. Chen. Network capacity of cognitive radio relay network. *Physical Communication*, 1:112–120, June 2008.

[63] X. Chen, Z. Zhao, T. Jiang, D. Grace, and H. Zhang. Inter-cluster connection in cognitive wireless mesh networks based on intelligent network coding. In *EURASIP Journal on Advances in Signal Processing*, 2009.

[64] Alfred Asterjadhi, Nicola Baldo, and Michele Zorzi. A distributed network coded control channel for multihop cognitive radio networks. In *IEEE Network*, volume 23, pages 26–32, 2009.

[65] Donglin Hu, Shiwen Mao, and Jeffrey H Reed. On video multicast in cognitive radio networks. In *IEEE INFOCOM*, 2009.

[66] Jin Jin et al. Multicast scheduling with cooperation and network coding in cognitive radio networks. In *IEEE INFOCOM*, 2010.

[67] W. Ren et al. Minimum-energy multicast tree in cognitive radio networks. *IEEE Asilomar Conference*, 2009.

[68] C. Xin et al. A novel layered graph model for topology formation and routing in dynamic spectrum access networks. In *IEEE Int. Symp. on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, 2005.

[69] I. Filippini, E. Ekici, and M. Cesana. Minimum maintenance cost routing in cognitive radio networks. In *IEEE 6th International Conference on Mobile Adhoc and Sensor Systems (MASS)*, pages 284 –293, October 2009.

[70] A. T. Hoang and Y.-C. Liang. Downlink channel assignment and power control for cognitive radio networks. *IEEE transactions on wireless communications*, 7(8), August 2008.

[71] M. Sharma, A. Sahoo, and K.D. Nayak. Channel selection under interference temperature model in multi-hop cognitive mesh networks. In *Second IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, pages 133–136, April 2007.

[72] H. Gossain, C. Cordeiro, D. Cavalcanti, and D.P. Agrawal. The deafness problems and solutions in wireless ad hoc networks using directional antennas. *IEEE Global Telecommunications Conference (GlobeCom) Workshops*, pages 108–113, Nov.-3 Dec. 2004.

[73] Y.R. Kondareddy and P. Agrawal. Synchronized mac protocol for multi-hop cognitive radio networks. In *IEEE International Conference on Communications (ICC)*, pages 3198 –3202, May 2008.

[74] Ivana Maric and Roy D. Yates. Cooperative multicast for maximum network lifetime. *IEEE JSAC*, 23(1):127 – 135, 2005.

[75] Tehuang Liu and Wanjiun Liao. Multicast routing in multi-radio multi-channel wireless mesh networks. *IEEE Transactions on Wireless Communications*, 9(10):3031 –3039, October 2010.

[76] Hon Sun Chin et al. Bandwidth-guaranteed multicast in multi-channel multi-interface wireless mesh networks. In *IEEE International Conference on Communications (ICC)*, pages 1 –5, June 2009.

[77] Nguyen Hoang Lan and Nguyen Uyen Trang. Channel assignment for multicast in multi-channel multi-radio wireless mesh networks. *Wireless Communications and Mobile Computing*, 9:557–571, April 2009.

[78] Guokai Zeng et al. Multicast algorithms for multi-channel wireless mesh networks. In *IEEE International Conference on Network Protocols*, pages 1–10, October 2007.

[79] Guokai Zeng et al. Efficient multicast algorithms for multichannel wireless mesh networks. *IEEE Transactions on Parallel and Distributed Systems*, 21:86–99, 2010.

[80] V. Ramamurthi et al. Multicast capacity of multi-channel multihop wireless networks. In *IEEE Wireless Communications and Networking Conference*, pages 1 –6, April 2009.

[81] Ayaz Isazadeh and Mohsen Heydarian. Traffic distribution for end-to-end qos routing with multicast multichannel services. *The Journal of Supercomputing*, 52:47–81, 2010.

[82] Miao Ma and D.H.K. Tsang. Impact of channel heterogeneity on spectrum sharing in cognitive radio networks. In *IEEE International Conference on Communications (ICC)*, pages 2377 –2382, May 2008.

[83] V. Fodor, I. Glaropoulos, and L. Pescosolido. Detecting low-power primary signals via distributed sensing to support opportunistic spectrum access. *IEEE International Conference on Communications (ICC)*, 2009.

[84] B. Mercier, V. Fodor, et al. Sensor networks for cognitive radio: Theory and system design. *ICT Mobile Summit*, June 2008.

[85] B. Mercier, V. Fodor, et al. Sensor networks for cognitive radio: Theory and system design. *ICT Mobile Summit*, June 2008.

[86] Yongle Wu, Beibei Wang, K.J.R. Liu, and T.C. Clancy. Collusion-resistant multi-winner spectrum auction for cognitive radio networks. In *Global Telecommunications Conference, 2008. IEEE GLOBECOM 2008. IEEE*, pages 1–5, 30 2008-Dec. 4 2008.

[87] Gaurav S. Kasbekar and Saswati Sarkar. Spectrum auction framework for access allocation in cognitive radio networks. In *MobiHoc '09: Proceedings of the tenth ACM international symposium on Mobile ad hoc networking and computing*, pages 13–22. ACM, 2009.

[88] TCI International. Tci 715 spectrum monitoring system data specification. *http://www.tcibr.com/ufiles/File/715Webp.pdf*, 2010.

[89] P. Hansen and B. Jaumard. Algorithms for the maximum satisfiability problem. *Computing*, 44:279 –303, 1990.