DESIGN AND ANALYSIS OF ROUTING AND ASSOCIATED TRANSMISSION SCHEMES IN COGNITIVE RADIO NETWORKS

by

Changliang Zheng

Dissertation submitted in fulfilment of the requirements

for the degree of

DOCTOR OF PHILOSOPHY

Department of Electronic Engineering
Faculty of Science
Macquarie University
Sydney, Australia

December 2012

ABSTRACT

Cognitive radio networks (CRNs) have emerged as an exciting technology to improve spectrum utilization and provide more available bandwidth. In CRNs, cognitive radio (CR) users dynamically access the unused spectrum granted to primary users. Such an opportunistic manner of spectrum access brings unique challenges for efficient communications between CR users. Much of the work on CRNs has focused on the PHY and MAC layer. However, routing design, as a key aspect of networking technologies, plays an important role in improving end-to-end performance for efficient communications between CR users in multi-hop CRNs. In CRNs, primary users' activities result in varying available channel sets for different CR users. Such spatial and temporal variations of channel (spectrum) availability have large impacts on routing design.

In this thesis, we first propose a multi-channel spectrum aware opportunistic routing algorithm for multi-radio CRNs with fast varying spectrum availability. In the proposed algorithm transmitters broadcast packets on multiple channels of CR links in order to reduce link delay and end-to-end delay. Additionally, the algorithm does not need a pre-setup route, and takes advantages of the broadcast nature of wireless transmission to overcome the unreliable link due to radio fading. The proposed algorithm delivers low delay and is able to support high traffic load.

We then propose a random linear coded scheme for efficient batch transmission over multiple channels between a single-hop communication pair in CRNs. The proposed scheme represents a key component in the network coding based opportunistic routing architecture of this thesis. We develop theoretical analysis and derive batch delays for the proposed scheme and two multi-channel ARQ based

schemes. We further extend the analysis to different batch transmission schemes, combined with different routing strategies, in a two-hop CRN. We derive network performance measures of the schemes in terms of batch delay, which provide insights into the data transmission capability in multi-channel CRN environment. The proposed scheme is less dependent on feedback channel and reduces batch delay significantly.

Finally, we consider a single transceiver multi-channel multi-hop CRN, which is a more practical scenario for low cost implementation. We propose a channel assignment and routing scheme, which minimizes the number of channel switches along a flow. We derive an efficient channel information dissemination algorithm that integrates the channel assignment scheme into the on-demand routing protocol and reduces the routing overhead. The proposed scheme achieves lower end-to-end delay, and higher and more stable throughput.

STATEMENT OF CANDIDATE

I certify that the work in this thesis has not previously been submitted for a degree nor has it been submitted as part of the requirements for a degree to any other university or institution other than Macquarie University.

I also certify that the thesis is an original piece of research and has been written by me.

In addition, I certify that all information sources and literature used are indicated in the thesis.

. . . .

Changliang Zheng

ACKNOWLEDGMENTS

I would like to express my sincerest gratitude to my supervisor, Prof. Eryk Dutkiewicz (Macquarie University, Australia), who has been always supportive whenever I met difficulties during my study. I appreciate his insights and suggestions that helped deepen my knowledge and shape my research skills. I would also like to express my greatest appreciation to my co-supervisor Dr. Ren Ping Liu (CSIRO ICT Centre, Australia), who has provided many inspirational instructions and discussions on research challenges. I thank him for his encouragement and stimulating suggestions in my research, and also for his help and care in my life. Without their remarkable insights, great knowledge, invaluable guidance and constant encouragements, I could not have finished this thesis.

I want to express my special thanks to my supervisor Prof. Zheng Zhou (Bejing University of Posts and Telecommunications, China), who introduced me into the scientific research area and provided his support and advice. I am thankful to my co-supervisor Prof. Rein Vesilo (Macquarie University, Australia) for his advice and encouragement throughout this work.

I would like to thank Dr. Iain Collings and Dr. Xun Yang for their advice and help on my research when I studied at CSIRO ICT Centre, Australia.

I give my thanks to all the members of the Wireless Communications and Networking Laboratory, Macquarie University. I am especially grateful to Dr. Gengfa Fang, Yiwei Yu and Jin Lai for their discussions during my research. I would like to thank all my friends who have supported and encouraged me.

Finally, I am greatly indebted to my parents and my sister for their endless support, encouragement and love.

To My Parents

List of Figures

Contents

Chapter 1

Introduction

1.1 Motivation

Cognitive radio (CR) [1] has emerged as a promising technology to enhance spectrum utilization and provide more wireless access opportunities. Wireless communications have experienced a huge success since Marconi's first demonstration of radio transmission at the end of 19th century [2]. With the explosive growth of wireless applications, such as wireless local area network (WLAN) and the third-generation (3G) cellular systems, wireless communications have become an important part of our daily life. Such growth imposes high demands on spectrum resources. As a scarce resource in nature, radio spectrum is under the management of government bodies, who enforce fixed spectrum assignment policies. Currently, most of the spectrum has been allocated to existing wireless systems. There is little spectrum left for new emerging wireless applications. On the other hand, it is reported that most of the allocated spectrum experiences very low utilization [3].

Realizing the contradiction between spectrum scarcity and low utilization, cognitive radio networks (CRNs) [4] have been proposed to explore the underutilized spectrum under a dynamic spectrum access (DSA) paradigm. In CRNs, CR users, who have not been granted a licensed spectrum band, are allowed to conduct their communications over the unused spectrum of primary users, who have been granted licenses to use these spectrum bands. The condition of DSA is that CR users must not interfere with primary users and the CR users must release the spectrum whenever primary users come back. In such an opportunistic manner, CRNs is able to improve the efficiency of spectrum utilization and obtain extra communication opportunities.

When a CR receiver is out of the transmission range of a CR sender, a multi-hop CRN is necessary to achieve the end-to-end data transmission. However, most of the research work on CRNs has been focused on single-hop scenarios, dealing with the problems associated with the physical (PHY) and medium access control (MAC) layers [5]. How to select a series of forwarding node, i.e., routing design, plays an important role in improving the end-to-end network performance. There have been extensive routing protocols designed for traditional wireless networks, but they are not suitable for CRNs. Routing design at the network layer is still a less explored area in CRNs. A unique challenge for routing design in CRNs is the dynamics of spectrum availability, which makes CRNs significantly different from traditional wireless networks. In CRNs, the available spectrum for CR communications may change with time and location for different users. Therefore, routing design in CRNs needs to account for such dynamics of spectrum availability.

Recently, there have been some efforts on routing protocol design that take spectrum availability into account. However, some important issues are yet to be investigated further.

Firstly, the varying rate of spectrum availability should be taken into account when designing routing protocols. The spectrum availability for CR users depends on primary users' activities. If the primary users are highly active, the spectrum availability changes frequently. In such highly dynamic spectrum environment, the average duration of the continuous available spectrum period is much shorter than that of the communication sessions of the CR users. However, most of the existing work on CR routing is based on forwarding packets via a fixed pre-selected path, which assumes that the spectrum change is relatively slow compared to the CR communication periods. When working in a CRN with a highly dynamic spectrum environment these routing schemes often fail and/or incur huge overheads due to frequent route maintenance. Effective routing protocols that are adaptive to such highly dynamic spectrum environments are required in order to leverage the benefits of CRNs.

Secondly, efficient and reliable multi-channel batch transmission related to routing protocols should be investigated. Opportunistic routing is a promising routing paradigm for CRNs. In opportunistic routing packets are transmitted in batches (groups of packets). Hence batch transmission design and performance analysis are important for opportunistic routing protocols. In practice, even with successful access to the required channels, data transmission between CR users could still suffer from failures due to channel fading. The performance of batch transmission depends on the efficiency of transmitting and retransmitting packets over multiple channels in parallel.

Thirdly, a CR device is often equipped with a single transceiver for low cost and easy implementation. Channel switching is required for such devices to work in DSA environments. In such cases channel switching delay becomes a performance issue especially in multi-hop CRNs. It has been reported that switching over a large spectrum frequency gap may result in long channel switching delay, e.g. in the order of 1 ms for a 10 MHz step in the frequency range of 20MHz-3GHz [6]. In traditional non-CR multi-channel multi-hop wireless networks, flow based channel assignment and routing protocols [7, 8] are effective in reducing the number of channel switches where all the nodes in a flow are assigned to the same frequency channel. However, the flow based approach is infeasible in the CR environment, as a common channel may not exist for all nodes along the path of the flow. Moreover, in CRNs channel information usually is required to be shared among

CR users along a route. For some CRNs consisting of embedded and energy constraint nodes, it is preferred to exploit efficient channel information dissemination and simple metric computation.

1.2 Thesis Contributions

In this thesis, a spectrum aware opportunistic routing algorithm is proposed and evaluated for highly dynamic CRNs. Related multi-channel transmission schemes are analysed. In addition, a maximum flow segment based channel assignment and routing algorithm is designed. More specially, the major contributions of this thesis are summarized as follows.

1. A multi-channel opportunistic routing algorithm is designed for CRNs under fast variation of spectrum availability. The proposed routing algorithm does not depend on any predetermined routing path, and packet forwarding decisions are made dynamically by the forwarding candidates who have successfully received the packets. The benefits of multiple channels are exploited for transmission. Packet overhearing is exploited in the proposed algorithm to accelerate data transmission by taking advantage of the broadcast nature of the wireless channel, such that transmission opportunities are fully exploited. Furthermore, network coding is employed to control packet duplication and enhance throughput. (See Section 3.3 in Chapter 3)

2. The opportunistic nature of CR links is captured by three metrics, namely, channel availability which depends on primary user activity, successful transmission rate which is influenced by channel fading, and channel access successful probability which is based on the contention level among the neighbouring nodes. For channel access successful probability, both accurate and approximate expressions are provided, depending on traffic and channel information of neighbours that a node owns. (See Section 3.2 in Chapter 3)

3. Traffic over a CR link composed of multiple channels is modelled as an M/Geo/1

queue. First, the service time of a packet over a single channel is analysed according to the packet transmission process by the three steps determined by the opportunistic nature of CR links. Then based on single channel analysis, the service time of a packet over a CR link consisting of multiple channels is identified as a variable, which follows a geometric distribution. Considering the packet arrival process is a Poisson process, the traffic over a CR link is finally modelled an M/Geo/1 queue. Our analysis shows that the delay of the link with multiple channels is lower than that with only one single channel. (See Section 3.2 in Chapter 3)

4. Performance of the proposed multi-channel opportunistic routing is evaluated via simulations. Compared with existing work, the maximum supported offered load is increased significantly using the proposed algorithm. This means the proposed algorithm is more suitable to work in the CRNs with heavier traffic by exploiting the benefits of multiple channels. It is found that there is a trade-off between the improvement of the maximum supported offered load and the improvement of the efficiency of the extra added channels. The proposed routing algorithm achieves a low end-to-end delay. In addition, it is shown that, by exploiting multiple channels, the proposed algorithm can compensate for the lost performance due to lower channel availability. (See Section 3.4 in Chapter 3)

5. Batch transmission over multiple lossy channels in CRNs is investigated in Chapter 4 and Chapter 5, which is a key component in the network coding based opportunistic routing paradigm. A random linear coded multi-channel batch transmission scheme is proposed in a single-hop CRN under practical channel fading conditions. In terms of reliable data transmission, the coded scheme can be viewed as an alternative to retransmission but it is more efficient than multi-channel automatic repeat request (ARQ) based retransmission schemes. This is because the coded scheme can reduce packet duplicates over multiple available channels by blurring packet information and making coded packets equal from the perspective of packet information. In addition, the coded scheme requires

fewer acknowledgment (ACK) messages, which makes it less dependent on feedback channels. (See Section 4.2 in Chapter 4)

6. The analytical solutions for the batch delay associated with the proposed one-hop random linear coded multi-channel batch transmission scheme are derived. The analytical delay solutions are also derived for ARQ based schemes. First, a general form expression of the batch delay associated with the coded scheme and ARQ based schemes is derived based on common flow charts of the batch transmission process. Then related probabilities in the general form solution are derived under different batch transmission schemes. The derivation of the batch delay is further illustrated and verified with two detailed examples. (See Section 4.3 in Chapter 4)

7. Performance of the proposed one-hop random linear coded multi-channel batch transmission scheme is evaluated via simulation and analysis. The theoretical analysis for different batch transmission schemes is verified via extensive simulations with various probabilities of channel availability and various packet loss probabilities. The analytical results and simulation results show that the batch delay performance of the coded scheme outperforms that of the ARQ based schemes. (See Section 4.4 in Chapter 4)

8. The coded batch transmission scheme in Chapter 4 is further extended to a two-hop CRN, combined with an opportunistic routing strategy in Chapter 5. The proposed 2-hop multi-channel batch transmission scheme exploits both packet forwarding and overhearing for data transmission by broadcasting different coded packets over multiple dynamically available channels. (See Section 5.2 in Chapter 5)

9. The batch delay associated with the proposed 2-hop batch transmission scheme is derived using a simplified two-stage transmission process model. Both general cases, where the channel condition of the relaying link is better than that of the direct link, and special cases, where the channel condition of the relaying link is worse than that of the direct link, are considered. The analysis is further validated via simulation. The analytical

results show that the proposed scheme can reduce batch transmission delay significantly. Theoretical analysis of batch transmission provides insights into the understanding and design of opportunistic routing for CRNs. (See Section 5.3 and Section 5.4 in Chapter 5)

10. A maximum flow-segment based channel assignment and routing algorithm is proposed for single transceiver CRNs. The definition of the maximum flow-segment and its construction procedure are introduced, based on which a channel assignment approach is proposed. The proposed scheme minimizes the number of channel switches by selecting the available channels that connect the maximum number of nodes in a routing path, which can further reduce the end-to-end delay. (See Section 6.3 in Chapter 6)

11. The channel assignment method is integrated into an on demand routing algorithm with a simplified channel information dissemination algorithm. Instead of forwarding full channel information at every node to the destination and calculating the delay metric at the destination, the proposed algorithm only forwards partial channel information with simple processing at the forwarding nodes. The channel information dissemination algorithm is implemented via a K-tuple piggybacked with a route request (RREQ) message. The K-tuple records the information on the number of consecutive upstream nodes that each channel can connect, which is updated at every forwarding node. Then channel assignment can be performed from the destination to the source accompanying with the transmission of route reply (RREP) message. (See Section 6.3 in Chapter 6)

12. The proposed maximum flow-segment based channel assignment and routing algorithm is evaluated via extensive simulations. The simulation results show that the proposed scheme achieves a significant reduction of the end-to-end delay. In addition, the proposed scheme delivers a high and stable throughput under heavy load. (See Section 6.4 in Chapter 6)

1.3 Thesis Organization

The remainder of this thesis is organised as follows. Chapter 2 provides a background and literature review of the related work for the research in this thesis. Chapter 3 presents a spectrum aware opportunistic routing design in multi-channel CRNs with highly dynamic spectrum based on a queueing analysis of CR links. In Chapter 4, a random linear coding based multi-channel batch transmission scheme is proposed and the associated batch transmission delay is derived. Two-hop multi-channel batch transmission schemes under different routing strategies are investigated in Chapter 5. In Chapter 6, a maximum flow-segment based channel assignment and routing algorithm is proposed and evaluated for single transceiver CRNs. Finally, conclusions of this thesis are drawn and future work is discussed in Chapter 7.

1.4 List of Publications

1. Changliang Zheng, Ren Ping Liu, Xun Yang, Iain B. Collings, Zheng Zhou, and Eryk Dutkiewicz, "Maximum Flow-Segment Based Channel Assignment and Routing in Cognitive Radio Networks", IEEE 73rd Vehicular Technology Conference (VTC Spring), Budapest, May 2011

2. Changliang Zheng, Eryk Dutkiewicz, Ren Ping Liu, Rein Vesilo, Gengfa Fang, and Zheng Zhou, "Opportunistic Routing in Multi-Channel Cognitive Radio Networks", 11th International Symposium on Communications and Information Technologies (ISCIT), Hangzhou, Oct. 2011 (Awarded Best Paper Prize)

3. Changliang Zheng, Eryk Dutkiewicz, Ren Ping Liu, Rein Vesilo, and Zheng Zhou, "Efficient Network Coding Transmission in 2-Hop Multi-Channel Cognitive Radio Networks", 12th International Symposium on Communications and Information Technologies (ISCIT), Gold Coast, Oct. 2012

4. Jin Lai, Eryk Dutkiewicz, Ren Ping Liu, Rein Vesilo, and Changliang Zheng, "Dynamic Spectrum Access with Two Channel Sensing in Cognitive Radio Networks", IEEE International Conference on Communications (ICC), Ottawa, Jun. 2012

5. Changliang Zheng, Eryk Dutkiewicz, Ren Ping Liu, Rein Vesilo, and Zheng Zhou, "Efficient Data Transmission with Random Linear Coding in Multi-Channel Cognitive Radio Networks", IEEE Wireless Communications and Networking Conference (WCNC), Shanghai, April 2013

Chapter 2

Background

This chapter provides background and review of work reported in the literature relevant to this thesis. First, an overview of cognitive radio networks is given as a general background. Then opportunistic routing is introduced and typical opportunistic routing protocols are reviewed. After that, current research on routing design in cognitive radio networks is reviewed. Finally potential multi-channel transmission schemes are discussed.

2.1 Cognitive Radio Networks

2.1.1 Motivation for Cognitive Radio and Cognitive Radio Networks

Radio spectrum is an essential resource for wireless communications but unfortunately it is a limited resource in nature. With the rapid growth of wireless applications, spectrum bands become more and more crowded. The increasing demand of wireless connectivity necessitates more available spectrum bands. The scarcity of spectrum has become much more severe artificially due to the current fixed spectrum assignment policy. Currently, the spectrum is usually regulated by government agencies, such as the Federal Communications Commission (FCC), with a fixed spectrum assignment policy. The FCC frequency allocation chart shows that most of the spectrum has been allocated to existing wireless services [9,10]. It seems that the spectrum is in high utilization since there are overlapping allocations in most frequency bands. However, this is not the real case. Under the current static assignment policy, a piece of spectrum is assigned to a legacy (licensed) user on a long-term basis for large geographical regions [11, 12]. In the large regions, according to the policy, only the corresponding licensed user has the right to access and exploit the granted spectrum, while other users are not allowed to use it even when the spectrum is not being used by the licensed user. On the other hand, a large portion of the assigned spectrum bands are in low utilization, 15% for the average usage and 85% for peak usage, according to FCC's report [3]. Dynamic spectrum access (DSA) is a promising solution to improve spectrum utilization efficiency and cognitive radio is a key enabling technology to achieve DSA paradigm [4].

2.1.2 What are Cognitive Radio and Cognitive Radio Networks

Cognitive radio (CR) is defined as a type of radio that can change its transmitter parameters based on the interaction with the environment in which it operates [3]. The concept of cognitive radio was first introduced by J. Mitola III, who also coined the term, cognitive radio [1, 13]. The main focus in Mitola's early work on cognitive radio was on the radio knowledge representation language (RKRL) and how to enhance the flexibility of personal services via RKRL. In Haykin's seminal paper [14], the basic cognitive radio functionalities were discussed from the view of communications, signal processing and networking for the first time.

Cognitive radio, as indicated in the definition, has two important characteristics, cognitive capability and reconfigurability [4, 14].

Cognitive capability refers to the ability to sense and obtain information from the surrounding radio environment, where the interested information may be transmission frequency, bandwidth and power. Through this capability, cognitive radio users can identify the spectrum that is not used at a certain time and location by other users. Further, cognitive radio users can select the best available spectrum via analysing the spectrum information.

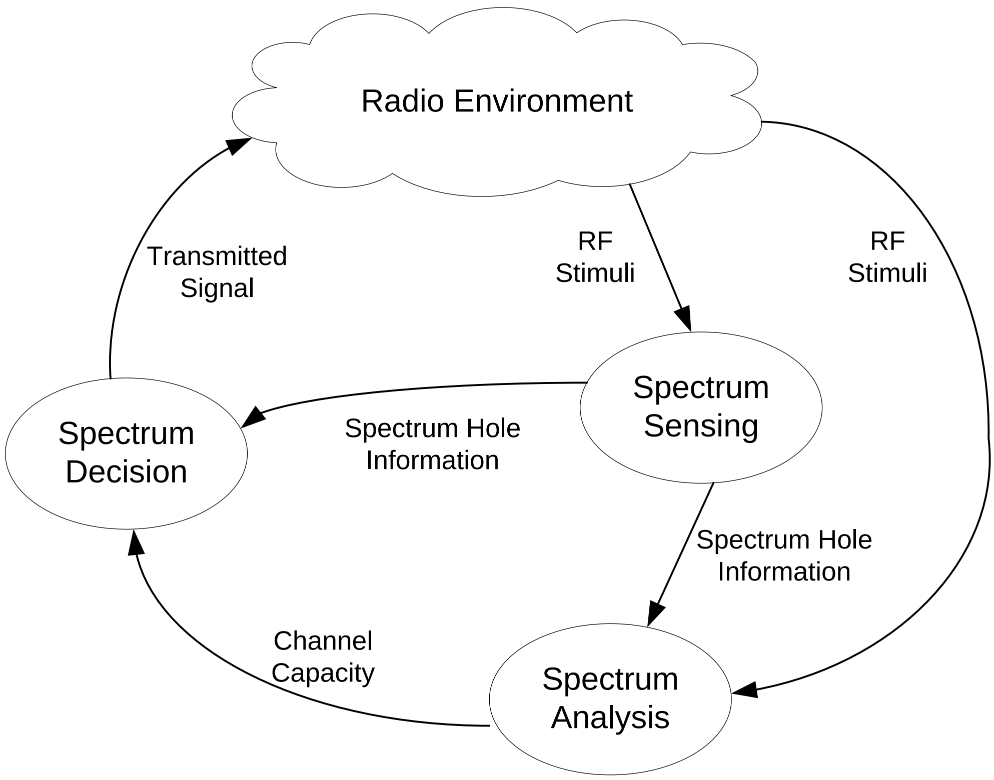


Figure 2.1: Cognitive cycle

The cognitive process for a cognitive radio to interact with the surrounding radio environment and adapt to the dynamic radio environment involves three basic cognitive tasks: spectrum sensing, spectrum analysis and spectrum decision. In the spectrum sensing task, a cognitive radio observes outside radio environment: monitors the available spectrum bands, captures the spectrum information, and detects the spectrum holes. A spectrum hole refers to a band of frequencies which is assigned to a primary user, but is not being utilized by that user at a particular time and specific geographic location [14]. In the spectrum analysis task, the characteristics of detected spectrum holes are analysed. For example, a cognitive radio estimates the channel-state information and predicts the capacity of the channel it can use. In the spectrum decision task, a cognitive radio selects the best spectrum and determines the modulation mode, the data rate and power for transmission. With the interaction with the surrounding radio environment, the three basic cognitive tasks construct a cognitive cycle as shown in Figure 2.1 [4].

Reconfigurability refers to the ability of a cognitive radio to dynamically change its operating parameters in real time according to the obtained information, through cognitive capability, about the radio environment. A general requirement on the reconfigurability is that the reconfiguration of operating parameters should not introduce any modifications to the hardware components, which is a similar requirement as imposed in software-defined radio. The reconfigurable parameters of a cognitive radio can be operating frequency, modulation scheme, transmission power and etc.

A *cognitive radio network* (CRN) is a wireless network that consists of cognitive radio users. Based on the cognitive capability and reconfigurability of CR users, CRNs are capable of working in a DSA mode to improve spectrum efficiency.

In the context of DSA [15], CRNs are also known as secondary networks while CR users are known as secondary users (SUs), since they are not granted the right to access and use licensed spectrum bands. In contrast, primary users are those users who are authorized with spectrum usage rights in licensed spectrum bands. In CRNs, CR users explore the spectrum in an opportunistic manner: they discover (sense) idle spectrum that are temporarily unused by primary users. CR users must vacate the spectrum whenever primary users reclaim it. With such a spectrum access mode, CR users must provide protection to primary users and there is no guarantee that spectrum will be available for CR users. These dynamics of spectrum availability are a unique characteristic of CRNs, which imposes major challenges on CRNs design.

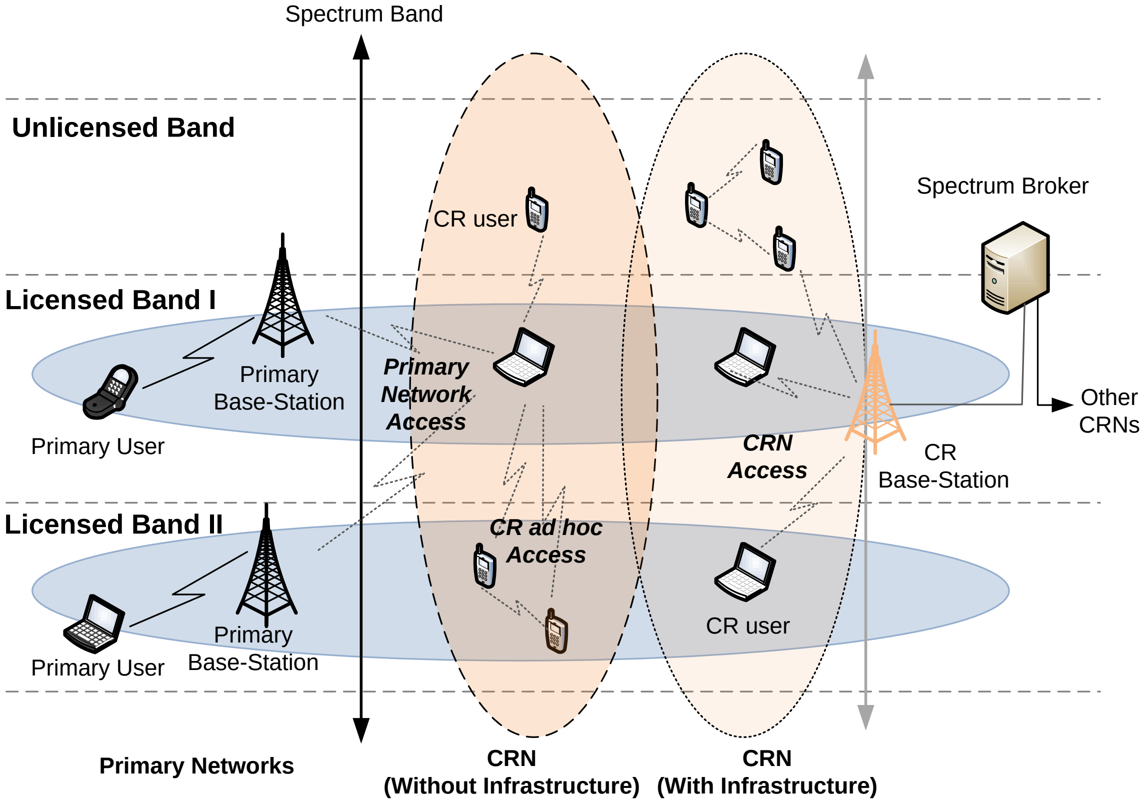


Figure 2.2: Cognitive radio network architecture

A general network architecture of CRNs is shown in Figure 2.2 [4]. There are two groups of components in the network architecture: primary networks and CRNs. Primary networks are composed of primary users and primary base stations, where a primary base station can provide access control and resource management. There are two types of deployments for CRNs: without infrastructure and with infrastructure. In the former deployment, CR users work in a distributed way. In the latter case, there is a CR base station to provide a centralized coordination between CR users in the network. The spectrum broker in the figure is used to coordinate spectrum sharing between different CRNs. Although CRNs can work in unlicensed spectrum bands, the promising advantages of CRNs come from their ability to dynamically access licensed bands. This attracts most of the current research on CRNs. To leverage the benefits of CRNs and enable its application in DSA, there are some open issues that need to be addressed.

2.1.3 Major Issues in CRNs

1. Spectrum sensing

Spectrum sensing is the task to get the information about the spectrum usage and existence of primary users in a geographical area [16]. This sensing task is the first step for CRNs. All the upper layer functions and protocols, such as MAC and routing protocols, should work based on the sensing results and be adaptive to spectrum changes [4].

The techniques for spectrum sensing can be classified into two groups: primary user receiver detection and primary user transmitter detection [12]. For primary user receiver detection based spectrum sensing, local oscillator detection [17] and proactive detection [18, 19] are two sensing methods. Local oscillator detection is designed for the communication systems like TV and radio broadcasts that have only one direction communication from the primary user transmitter to the primary user receiver. Proactive detection works for the communication systems where closed-loop control schemes, such as power control, are employed with feedback channels.

For primary user transmitter detection based spectrum sensing, matched filter detection, energy detection and feature detection are the three main sensing techniques [11,20]. Matched filter detection [21] is the optimal way for primary user signals detection in stationary Gaussian noise when a CR user has a priori knowledge of the primary user signals, since it maximizes received signal-to-noise ratio. Energy detection is the most common technique of spectrum sensing due to its simplicity and easy implementation [16, 22, 23]. When the noise power is known to CR users, energy detection is the optimal sensing scheme for unknown primary user signals [21]. Feature detection [24, 25] determines the presence of primary user signals based on the specific features associated with the received signals. Feature detection can overcome the impact of the uncertainty in noise power and it can also identify different signals from different networks.

All the spectrum sensing schemes mentioned above are individual local sensing performed by a single CR user. However, the sensing performance of such local sensing experiences degradation due to noise uncertainty, fading and shadowing effects. To solve the problem, cooperative sensing [11,26,27] is proposed by taking advantage of the spatial diversity and multiuser diversity. In cooperative sensing, multiple CR users work together to perform spectrum sensing and then the sensing information from different CR users is combined to make a more accurate final judgement on the presence of a primary user [16].

2. Spectrum sharing

Spectrum sharing refers to spectrum allocation and spectrum access, which is performed based on spectrum sensing results. The aim of spectrum sharing is to utilize the spectrum resources efficiently and fairly schedule spectrum among multiple CR users.

Spectrum sharing schemes can be categorized according to various criteria [4, 11]. Based on the spectrum bands that CR users exploit, there are open spectrum sharing and hierarchical access sharing approaches. Open spectrum sharing only use unlicensed spectrum bands, while in hierarchical access sharing CR users access licensed spectrum bands [28]. According to spectrum access techniques, they are divided into overlay spectrum sharing and underlay spectrum sharing. In overlay spectrum sharing schemes [29] , CR users only use the spectrum when primary users are not using it, while in underlay spectrum sharing [30] CR users and primary users can use the spectrum simultaneously but CR users are required to limit the interference to primary users. Based on the network architecture, the centralized schemes use a central control entity and the distributed schemes allow CR users act individually. Cooperative sharing schemes [31] and non-cooperative schemes [32] are classified depending on whether CR users make their decisions based on a common objective.

For spectrum allocation, optimization techniques and game theory are two major approaches. Optimization methods [33] work in a centralized manner and global information is often required. With game theory [34], spectrum allocation can be performed based on local information in a distributed manner but with some performance degradation.

Spectrum access determines the timing for CR users transmission and this is usually implemented by various CR MAC designs. Considering spectrum access, CR MAC protocols employ three techniques: random access, slotted and the combination of both [25]. More information on CR MAC can be found in [35].

3. Routing in CRNs

At the network layer, routing is the major issue which is still less explored to provide multi-hop data transmission in CRNs to improve network performance.

The dynamic and opportunistic spectrum access manner of CR users imposes unique challenges on routing design in CRNs. The problem of routing design in multi-hop CRNs is not only to establish and maintain wireless multi-paths by choosing the forwarding nodes, but also to consider spectrum selection for the links along the path [36]. In other words, routing design is closely coupled with spectrum selection in multi-hop CRNs. As a result, available spectrum information needs to be disseminated at least between neighbouring nodes, or beyond that, to guarantee the point-to-point and end-to-end connectivity [5]. Moreover, the level of primary users activity determines the timescale of the period that the spectrum is continuously available [37], which must be accounted for in routing design.

Routing design in CRNs is the main topic of this thesis and an extensive review on the current research work will be provided in following sections

2.2 Opportunistic Routing

Opportunistic routing [38–41] is a recently emerging and promising routing mechanism for multi-hop wireless networks. This routing technique takes advantages of the broadcast characteristic of wireless transmission. By dynamically selecting best forwarding nodes, opportunistic routing can overcome the unreliable links introduced by the dynamic wireless environment and achieve a significant improvement in end-to-end performance in multi-hop wireless networks. Both theoretical analysis [42–46] and experimental evaluation [38, 39, 47] have confirmed the potential benefits of opportunistic routing in various traditional wireless networks. Such advantages further inspired some initial explorations of opportunistic routing's application in routing protocol design in CRNs with dynamic spectrum environments [48, 49].

2.2.1 Basics of Opportunistic Routing

The broadcast nature is a unique characteristic of the wireless medium. A unicast packet transmitted to a specific node in wireless networks is actually broadcast in the sender's neighbourhood and all the one-hop neighbours have opportunities to receive the packet.

However, in traditional routing protocol design [50–53], such broadcast nature of wireless transmission is not exploited. Broadcast is even considered as a drawback producing interference to the neighbouring nodes other than the specific destination node. Traditional routing protocols forward data packets between a given communication pair along a fixed best routing path, which is selected before data transmission starts according to various routing metrics, such as hop counts and link quality. Traditional routing mechanisms for wireless networks follow the routing concept in wired networks by abstracting wireless links as point-to-point wired links. In traditional routing, a node will discard any overheard packets not destined to itself, even when the packets are received correctly. In fact, these overheard packets could be helpful for packet forwarding and avoiding unnecessary retransmissions when the specific relay node fails to receive the packets.

Opportunistic routing schemes, in contrast, fully utilize the broadcast nature of the wireless medium. An important concept in opportunistic routing is to explore packet overhearing associated with wireless transmission. The basic idea behind opportunistic routing is the coordination between the network layer and the MAC layer in a way that a set of relay candidates are selected at the network layer and one best relay node is chosen at each transmission at the MAC layer. More specially, in opportunistic routing, the source node or a relay node selects a subset of its one-hop neighbouring nodes as relay candidates. The relay candidates are selected and ordered according to some metrics e.g. their distance to destination node. Then the packet holding node broadcasts the packet. Among the relay candidates who receive the packet, the one associated with the highest order is chosen as the actual relay node to forward the packet. The process will be performed until the destination node receives the packet. In such a manner, the overheard packets of relay candidates are potentially used for packet forwarding in opportunistic routing, rather than be dropped as in traditional routing.

Opportunistic routing enjoys two advantages via exploring packet overhearing [41,54]. The first advantage is that opportunistic routing can increase transmission reliability. A transmission can be received independently with a probability by each relay candidate. Thus the probabilities of transmission failure and retransmission are reduced with opportunistic routing. This can be illustrated via an example. In Figure 2.3, node src is the source node and node dst is the destination node. Nodes A, B and C can help to forward packets from src to dst. Every link has a packet delivery probability as shown in the figure. With a traditional routing scheme, a specific node is selected as the relay node, say B, and builds a fixed route, src-B-dst, for packet delivery from src to dst. The end-to-end delivery probability over this fixed route is assumed to be 20%. When a transmission from src to B fails, a retransmission is required. With opportunistic routing, nodes A, B and C are all relay candidates, and the specific relay node for a packet is selected after the transmission from src. As the transmission in wireless networks is essentially a broadcast, nodes A and C have a chance to receive the packet even if B fails to receive it. Then either A or C is selected as the relay node to forward the packet, which can successfully avoid the retransmission from src to dst. Compared to a fixed route, src-B-dst, selected by a traditional routing scheme, any of the possible routes, src-A-dst, src-B-dst and src-C-dst, can be dynamically employed to deliver a packet from src to dst. Therefore the end-to-end delivery probability with opportunistic routing is $1 - (1 - 20\%)^3 = 48.8\%$, improved by 28.8% compared to that of traditional routing.

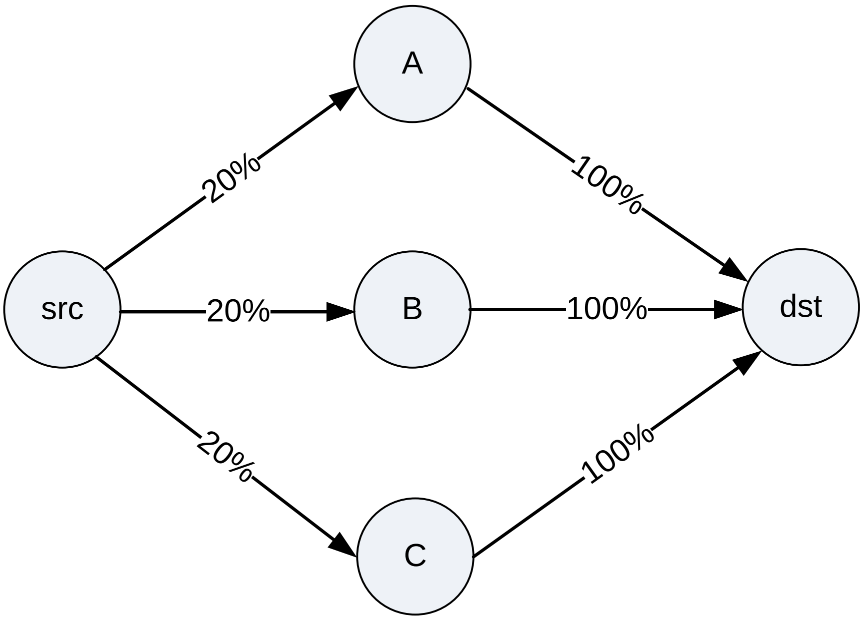


Figure 2.3: Transmission reliability enhancement with opportunistic routing

The second advantage is that opportunistic routing can enhance the transmission range. In opportunistic routing, not only good quality short range links but also poor quality long range links are exploited. Therefore a packet can be transmitted to the farthest relay within a transmission. This benefit can be well illustrated in a chain topology network as in Figure 2.4. In the network, different range links have different link qualities in terms of packet delivery probability: short range links such as (src, A) with a 80% delivery probability, medium range links such as (src, B) with a 60% delivery probability, and long range links such as (src, C) with a 40% delivery probability. With opportunistic routing, a transmission of a packet at node src may be received by node C over a long distance. The packet could be delivered to the destination with only two transmissions. However, good quality links are usually selected for the best routing path by a traditional routing scheme and thus it requires at least four transmissions to deliver a packet from src to dst.

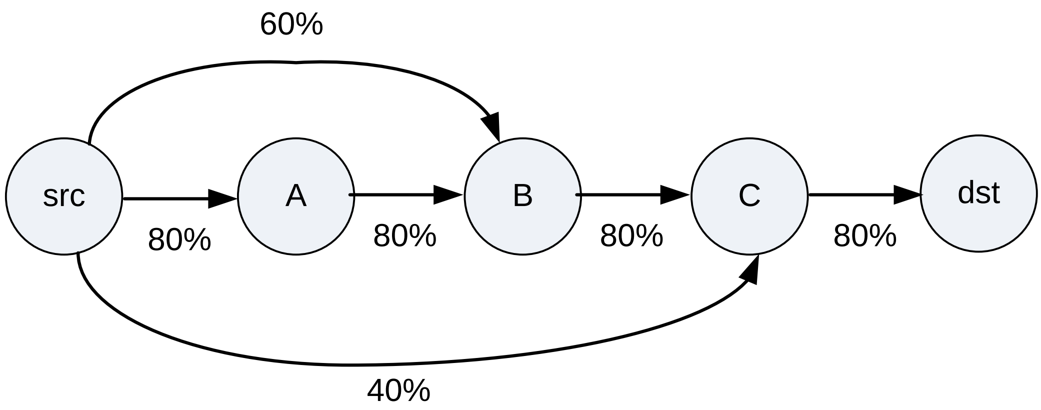


Figure 2.4: Transmission range improvement with opportunistic routing

Opportunistic routing has two main components: relay candidate set selection and coordination method [40, 41, 55]. In relay candidate set selection, a set of next-hop relay candidates is selected for a sender to forward a packet to the destination node. Generally, a node can be selected into the relay candidate set if it is a direct neighbouring node of the sender or it is closer to the destination node than the sender in terms of a specific routing metric. In addition, relay candidate set selection schemes determine and arrange these relay candidates in the candidate set in order of priority based on some criteria. The metrics used to prioritize relay candidates include expected transmission count (ETX) [38, 39], hop count [56, 57], geo-distance [58] and so on.

A coordination method refers to the selection of the best relay candidate as the actual relay, who forwards the packet and stops other relay candidates from forwarding this packet, when a packet is broadcast. For a sender, the coordination is employed to choose the best candidate to forward packets, while, from the relay candidate's point of view, the coordination is used to make a decision whether the candidate forwards the overheard packets. There are two kinds of typical existing coordination methods: timer-based coordination and network coding.

In a timer-based coordination method, all the relay candidates are ordered according to a metric. The information on the candidate order is carried in a packet header. After the data packet is broadcast, each relay candidate responds according to the predetermined priority order in its corresponding time slot. For example, the candidate with i^{th} priority responds at the i^{th} time slot. A candidate responds only when no other candidates with higher priority respond. Once a candidate responds, it is selected as the actual relay node to forward the current packet and all the candidates with lower priority will not forward the packet.

Network coding actually bypasses the requirement of coordination between relay candidates. A coordination method in opportunistic routing is used to avoid packet duplication but working with network coding, there are no duplicate packets. In network coding based opportunistic routing, a data flow from the source to the destination is divided into batches to code and decode. A batch includes several original packets without coding. The source broadcast random linear combinations of the original packets. When a relay node forwards a packet, it broadcasts a random linear combination of all the packets already received and stored in its buffer. In such a way, all the packets transmitted by different nodes are different. At the destination, the original packets are restored from the coded packets via Gaussian elimination.

2.2.2 Review of Typical Opportunistic Routing Protocols

The Extremely Opportunistic Routing (ExOR) was proposed by Biswas and Morris in [54], which was then enhanced and implemented on the RoofNet testbed at MIT [38]. ExOR is the first opportunistic routing scheme focusing on performance improvement in wireless network.

ExOR selects forwarder candidates based on ETX and coordinates their transmission with a timer-based method. ExOR broadcasts a packet without a predetermined next hop forwarder, the choice of which is deferred until this transmission is completed. Then the forwarder is selected among the nodes who successfully receive the packet. This selection is based on which node is closest to the destination, in terms of a metric called ETX. After that the selected node forwards the packet to the next hop in a similar way. Accordingly, the operation of ExOR is divided into three stages: selecting the forwarding candidates, acknowledging transmissions, and deciding whether to forward a received packet. As mentioned before, neighbouring nodes of a sending node that are closer to the destination are selected as forwarding candidates and they are prioritized according to ETX. In order to reduce duplicate transmissions, a slotted acknowledgement mechanism is used to coordinate the transmission of forwarding candidates. An acknowledgement message not only reports the successful reception of a packet, but also carries the information on which successful recipient is with the highest priority known to the sender of the ACK. All the nodes need to listen to the whole ACK window, which is used to suppress duplicate forwarding. The slotted acknowledging mechanism is shown in Figure 2.5. To reduce the coordination overhead, packets are transmitted in batches. With this method, the source node broadcasts a batch of packets and then other nodes are coordinated to forward the packets in this batch. The source node starts to transmit the next batch of packets after it completes the transmission of the current batch of packets.

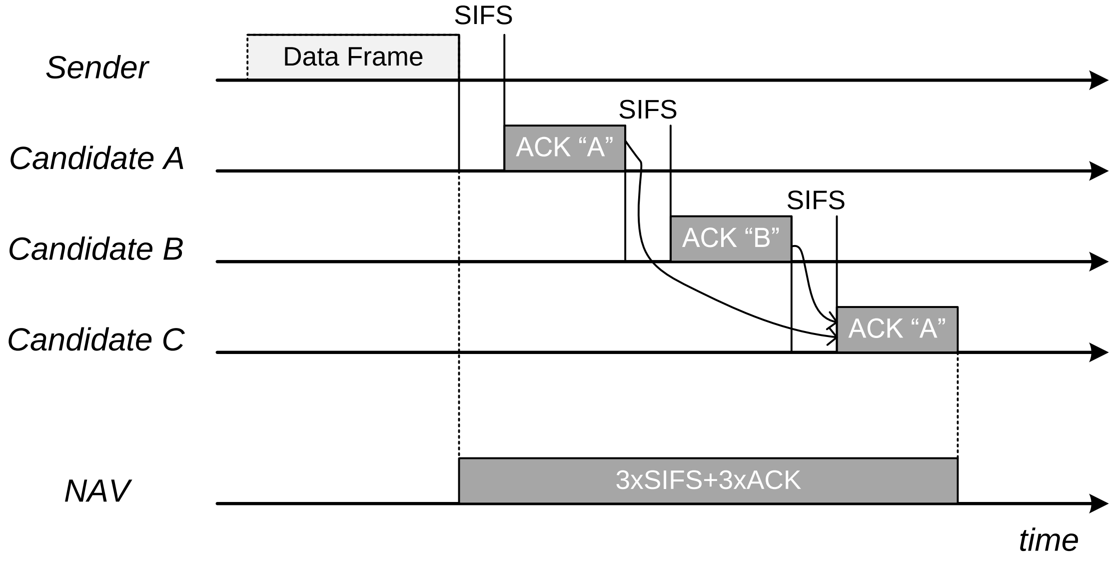


Figure 2.5: Acknowledgment sequence in ExOR

A main achievement of ExOR is to demonstrate that opportunistic routing outperforms traditional routing based on real measurements in a testbed. However, ExOR uses a global scheduler for coordination and thus loses the spatial reuse.

ExOR is a seminal opportunistic routing protocol. By showing the effectiveness of the concept of opportunistic forwarding and performance gain over traditional routing approaches, ExOR has encouraged much work in the area of opportunistic routing. Researchers attempt to design more efficient schemes to improve protocol performance by optimizing forwarder candidate selection and forwarding coordination. In [47], the authors proposed a simple opportunistic adaptive routing protocol (SOAR) to achieve high throughput and fairness, which combines adaptive forwarding path selection, priority timer-based forwarding, local loss recovery and adaptive control. Zeng et al introduced a new forwarder selection metric named expected one-hop throughput (EOT) to balance packet advancement and medium time delay in geographic opportunistic routing [59].

In [60], a new metric EAX (expected any-path transmissions time) was defined which captures the expected number of any-path transmissions needed to successfully deliver a packet between two nodes. A routing metric EATT (expected number of anypath transmissions time) and a SMAF (shortest multirate anypath first) algorithm were proposed to integrate opportunistic routing and multiple transmission rates [61]. These opportunistic routing schemes all employ timer based coordination, but some of them have not been limited to a global scheduler as in ExOR.

The MAC-independent Opportunistic Routing and Encoding Protocol (MORE) was proposed in [39], combining opportunistic routing and intra-flow network coding. MORE is the first work to exploit network coding as the coordination method among forwarding nodes, which avoids the use of a global scheduler in ExOR. In MORE, packets are randomly mixed before being forwarded, which guarantees that forwarding nodes that receive the same transmission do not forward the same packet.

At the source node, packets are first divided into batches. A batch consists of several native packets (i.e., uncoded packeted). The source keeps generating random linear combinations of the native packets in the same batch and broadcasting the coded packets. When the source receives an ACK for this batch from the destination, it will stop the transmission of the current batch and moves to next batch. In MORE, data packets are always coded. A code vector, consisting of the combination coefficients, is used to indicate how a coded packet is created from the native packets. For example, a coded packet can be expressed as $p_j^* = \sum_i c_{ji} p_i$, where c_{ji} is the random coefficient, and p_i is a native packet in the current batch. The code vector of coded packet p_j^* is $c_j = (c_{j1}, ..., c_{ji}, ..., c_{jK})$, where K is the number of native packets in the current batch. A code vector and a forwarder list are carried with a coded packet.

When a forwarding node receives a coded packet, it checks if it is in the forwarder list. If so, the node further checks for the innovativeness of the packet (i.e., if the packet is linearly independent from the previously received packets at this node). The forwarding node only stores innovative packets. The reception of an innovative packet triggers the forwarding node creating and broadcasting a new coded packet, which is a random linear combination of the coded packets the node has heard and stored in its buffer. In fact, a linear combination of coded packets is still a linear combination of native packets. This can be illustrated as $p^{**} = \sum_j r_j p_j^* = \sum_j (r_j \sum_i c_{ji} p_i) = \sum_i (\sum_j r_j c_{ji}) p_i$. The code vector for the recoded packet consists of $\sum_j r_j c_{ji}$, where $i = 1, ..., K$.

The destination also checks if the received packet is innovative. When the destination has received K innovative packets, it restores the native packets in the batch by doing a simple matrix inversion and sends an ACK to the source.

Experimental results based on testbeds show that MORE produces a significant performance gain over ExOR.

MORE is one of the most classical opportunistic routing, based on which much research work has been conducted to extend the idea and improve the performance of network coding based opportunistic routing. In [62] , CodeOR was proposed to enhance the throughput of MORE by transmitting a window of multiple segments concurrently. Lin et al further proposed a new protocol SlideOR [63], which encodes source packets in overlapping sliding windows such that coded packets from one window position may be useful towards decoding the source packets inside another window position. A novel cumulative coded acknowledgment scheme was designed in CCACK [64, 65], which allows nodes to acknowledge network-coded traffic to their upstream nodes in a simple way. Working in such a manner, CCACK greatly improves network performance in terms of throughput and fairness. In addition to these single flow based opportunistic routing schemes, Radunovic et al proposed an optimization framework based on network utility maximization to solve multiple flows, fairness and scheduling problems in network coding based opportunistic routing protocols [66].

Opportunistic routing has shown its benefits over traditional routing in recent research above by exploiting the broadcast nature of the wireless medium. In opportunistic routing, forwarding nodes can be selected after transmissions, which means that the selection can be conducted based on the transmission results and the actual network environment. With such a dynamic forwarder selection way, opportunistic routing is more suitable for networks in dynamic environments, such as CRNs with DSA. This advantage has attracted some research on the application of opportunistic routing in CRNs [48, 49] and we will provide further review of this work in the next section.

2.3 Routing in CRNs

Routing is an essential issue to be investigated when dealing with multi-hop cognitive radio networks, which plays a key role in improving end-to-end network performance. Routing protocols have been extensively studied in traditional wireless networks. However, those routing protocols are not directly applicable to work in CRNs due to the differences between CRNs and the traditional networks.

A unique challenge for routing design in CRNs is the dynamics of spectrum availability, which is highly influenced by primary users' activities. Research work in CRNs has been mainly focused on the physical layer and MAC layer, such as spectrum sensing and spectrum sharing. Only recently, routing design in multi-hop CRNs has attracted more interests from the research community. The existing CR routing protocols can be classified according to different criteria.

Based on the scope of spectrum knowledge [36], CR routing can be categorized into two main groups:

- Routing solutions with full spectrum knowledge

- Routing solutions with local spectrum knowledge

Based on their supporting functionalities [25], CR routing can be divided into three main classes:

- Routing with spectrum decision
- Routing with joint spectrum decision and primary user awareness
- Routing with joint spectrum decision and re-configurability.

Based on the relative timescale of the duration that the licensed spectrum is available for CR users [37], CR routing can be classified into three main categories:

- Routing with static spectrum
- Routing with dynamic spectrum
- Routing with highly dynamic spectrum

The different classes of CR routing protocols are summarized in Figure 2.6.

Classification Metrics	Scope of spectrum knowledge	Supporting functionalities	Relative timescale of spectrum duration
Categories	1. Routing with full spectrum knowledge 2. Routing with local spectrum knowledge	1. Routing with spectrum decision 2. Routing with joint spectrum decision and primary user awareness 3. Routing with joint spectrum decision and re-configurability	1. Routing with static spectrum 2. Routing with dynamic spectrum 3. Routing with highly dynamic spectrum

Figure 2.6: Classes of CR routing protocols

In the following subsections, we will review the current research on CR routing design based on the varying timescale of spectrum availability. This classification can provide more insights into how routing design in CRNs is influenced by primary users' activities and which environment a specific routing protocol is applicable to.

2.3.1 Routing with Static Spectrum

In CRNs, spectrum availability depends on primary user's behaviour. Routing with static spectrum refers to the routing schemes that work in such a spectrum environment, where the spectrum availability changes very slowly. More specifically, the continuously available time of the spectrum for CR users is longer than the communication flow duration. In such a scenario, an available channel (spectrum) is considered as a permanent resource during the communication, from CR users' point of view. Therefore, CR routing with static spectrum does not need to consider channel availability varying with time but only consider different available channels of CR nodes along a route.

In [67–70], graph-based approaches were employed for routing design with static spectrum. In [67,68], the authors proposed a layered graph model to deal with channel assignment and routing jointly in CRNs. The general design process of these routing protocols consists of graph abstraction and route calculation. First a logical graph is constructed to describe the physical network topology and then a path connecting the source node and the destination node is computed. A layered graph is constructed based on the available channel information of every CR node in the network, which is gathered by a central server. If there are totally N channels, then N layers are constructed in the graph, each layer corresponding to a channel. Each node and N additional subnodes are denoted as vertices of the graph, with one subnode each layer. There are three types of edges in the layered graph: access, horizontal, and vertical edges. Access edges are used to connect a node with all the corresponding subnodes. A horizontal edge is added between two subnodes in the same layer if the corresponding channel associated with the layer is available for the two nodes and they are in the communication range of each other. Vertical edges connect the subnodes of the same node. As an example, Figure 2.7 [68] illustrates a chain CRN and the corresponding layered graph. Before calculating the routing path, the edge weights should be assigned. The weights of the horizontal edges usually account for the specific quality of the CR link, while the weight of the vertical edges can be set relative to channel switching overhead. Then the path between the source node and the destination node can be obtained across the layered graph and consequently the channel assignment and routing problems can be solved jointly.

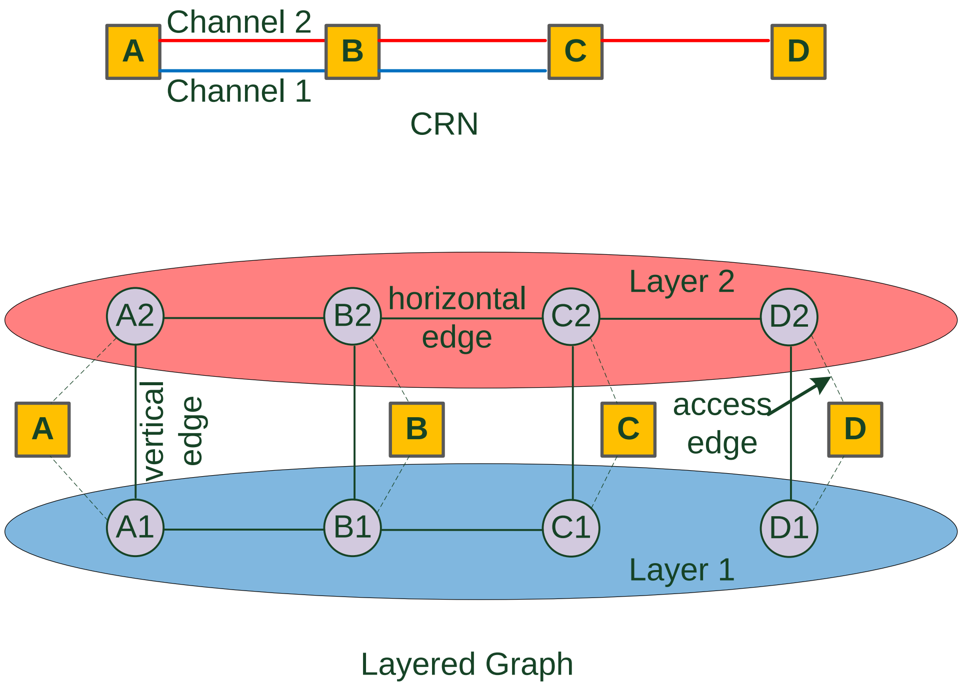


Figure 2.7: A chain CRN and the corresponding layered graph

In [69], a similar method was used to address channel assignment and routing in static CRNs. A coloured graph is constructed to represent the physical network topology, instead of a layered graph. In the coloured graph, different colours denote different channels in the network. Then a path connecting the source node and the destination node is calculated with a centralized iterative algorithm. With the coloured graph, a major concern to assess the path in the setting up stage is the interference among links. The authors of [70] proposed a collaborative route and spectrum selection scheme using conflict graphs. The scheme finds all possible candidate routes and all possible channel assignments for each route to obtain a best route and channel combination. In the process, a conflict graph is constructed to schedule a conflict-free channel assignment.

These graph-based schemes can provide general frameworks to jointly find a route and assignment channels. However, they usually need to gather all the topology and channel information in the network, which introduces large signalling overhead. A central entity is used to perform the function of graph abstraction and path computation with complex algorithms.

Another group of routing protocols [71–74] for static spectrum CRNs are designed based on some delay metrics introduced by CR nodes along routing path. In these protocols, a route and channel assignment with minimum delay is the final selection. In [71,72], both the switching delay between frequency bands and backoff delay caused by multiple users' contention in the same band are taken into account. Moreover, a multi-flow scheduling scheme is designed to avoid frequent frequency bands switching among flows. The delay metric at a node is divided into two parts, i.e. node delay and path delay. For the node delay, switching delays among multiple flows traversing the node and backoff delays due to neighbours contention are accounted. For the path delay, the proposed scheme also considers two components. One component is the intra-flow switching delay, while the other component is the backoff delay introduced by multiple consecutive nodes sharing an identical frequency band. Then the cumulative delay metric is calculated by adding the delay metric at each node along the routing path. Finally an optimal route combined channel assignment is picked out based on the cumulative delay. In [73, 74] a queueing delay was integrated into the routing metric in addition to the switching delay and the backoff delay in previous works, which makes the new metric more effective and practical to evaluate candidate routes. Furthermore, a local coordination mechanism [74] was proposed to balance flow traffic in the neighbourhood of an intersecting node.

These delay metric based routing protocols do not need a central entity to perform path calculation and channel assignment as in graph-based schemes. They can hence work in a distributed way. A drawback of these schemes is that all channel information at each node along a routing path is required to be disseminated to the destination for cumulative delay calculation. In [75], we designed a maximum flow-segment based channel assignment and routing algorithm. The proposed algorithm can significantly reduce channel switching with more efficient channel information dissemination mechanism and less computation. We will discuss this work in Chapter 6.

2.3.2 Routing with Dynamic Spectrum

Routing with dynamic spectrum refers to the routing protocols in such a CRN, where the holding time of the spectrum by CR users is comparable to the communication flow duration. Spectrum availability may change during the communication period, but not frequently. In such a dynamic environment, potential communication interruption caused by primary users presence is a major concern in routing. This challenge is usually addressed in existing work via two methods: 1) choosing a stable route based on the statistical characteristics of the spectrum and 2) providing route recovery and maintenance mechanisms.

A spectrum-tree based on-demand routing protocol (STOD-RP) was proposed by Zhu et al [76] to collaboratively deal with the route and spectrum selection in a dynamic spectrum CRN. In STOD-RP, a new routing metric by considering both the route quality and the spectrum availability is defined as

$$C_i = [O_{ca} + O_p + \frac{P_{kt}}{r_i}]\frac{1}{1 - e_{pti}} \bullet \frac{1}{T_{l_i}} \tag{2.1}$$

where O_{ca} is the channel access overhead, O_p is the protocol overhead and P_{kt} is the size of packet. These three parameters are constants for a specific technology. r_i and e_{pti} are the link rate and the packet error rate, respectively. These five parameters reflect the resource consumption of a link. The key CR parameter in the equation is T_{l_i}, which is the time duration that a spectrum band is available to the link. T_{l_i} is an indicator of the link stability with regard to primary users' behaviour and it can be predicted from the statistical history of primary users' activities. This new defined routing metric is a key factor to enable STOD-RP to work in a dynamic spectrum environment. STOD-RP builds a spectrum-tree in each available spectrum band. The root in each spectrum records the basic information about the spectrum-tree topology, like the routes to other non-root nodes. Another important type of nodes are the overlapping nodes, which are shared by multiple spectrum-trees and can work in multiple spectrum-trees simultaneously. STOD-RP combines tree-based proactive routing with on-demand route discovery. A source node first send a message to the root node to enquire about the frequency band of the destination using a recorded proactive route and then starts route discovery in an on-demand manner accordingly. Another key feature of STOD-RP is a spectrum-adaptive route recovery mechanism, which is used to deal with link failures cased by primary user activities. When a spectrum band becomes not available, both spectrum handoff and path rerouting will be triggered for route recovery. A limitation of this work is that the network architecture is assumed to be mesh networks arranged in a tree hierarchy.

In [77], a distributed CR routing protocol for ad hoc networks (CRP) was proposed with explicit protection for primary user receivers. In addition, CRP selects spectrum with consideration of link stability and offers route maintenance. The route setup in CRP takes two steps. In the first step, the best spectrum band is identified by each CR user based on local environmental observations. An optimization function is designed to serve as a measure for the initiative displayed by the CR user for participating in the route. In the second step, the initiative is mapped to a delay function for forwarding the RREQ message. This delay provides a ranking of candidate forwarding nodes by allowing favourite nodes to forward the RREQ message earlier. When making spectrum selection, the probability of bandwidth availability is used. This metric provides an evaluation for the candidate CR users if the chosen spectrum band can probabilistically guarantee the bandwidth availability to avoid frequent spectrum band switching due to primary user's activity. Another important metric is the overlap area between the transmission coverage of the primary user transmitter and the CR user, which is shown as the gridded areas in Figure 2.8. Primary user receivers may be present in the region of the overlap, and thus the overlap area must be minimized at each forwarding CR user in the selected route to protect primary user receivers. As for the route maintenance, a CR user notifies its previous hop node to discover a new route if the CR user moves too close to primary user transmitters. CRP is effective to work with dynamic spectrum and provides primary user receiver protection. On the down side, in CRP geographic information of primary user transmitters is assumed to be known to CR users.

The authors in [78] proposed a spectrum aware routing protocol based on geographic, SEARCH, which adapts to dynamic spectrum in CRNs by providing primary user activity awareness and route maintenance. SPEAR [79] employs local adaptation to deal with channel availability variation due to the actions of primary users. In [80], a probabilistic metric was exploited to ensure routes stability and availability. Similar schemes include [81–84]. In summary, to adapt to the dynamic spectrum in CRNs, route stability based on spectrum statistics and route maintenance are two effective methods.

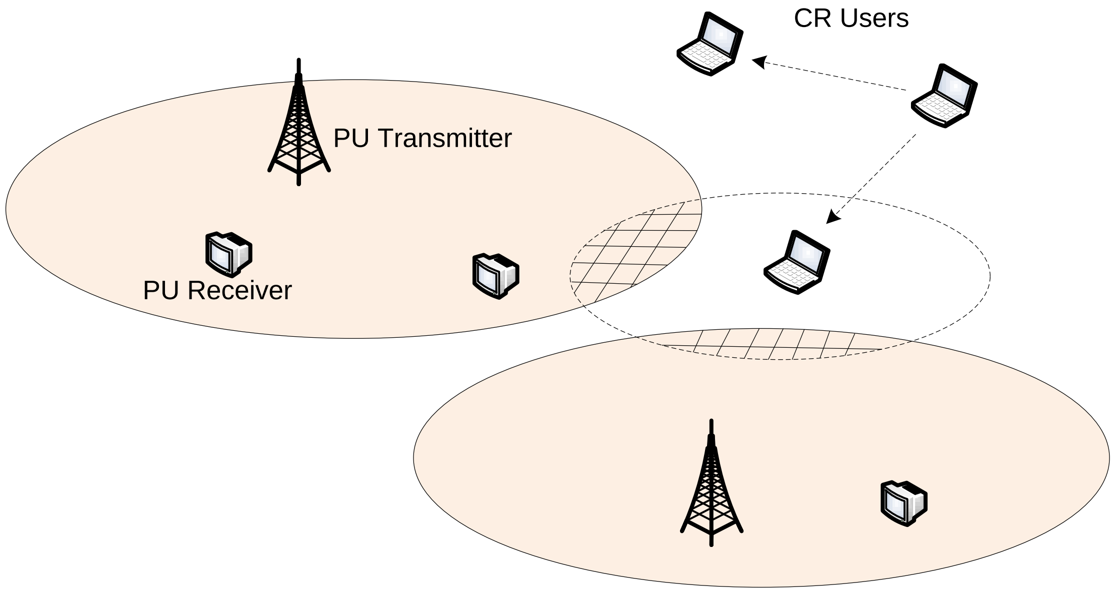


Figure 2.8: Forwarding node selection with primary user (PU) receiver protection

2.3.3 Routing with Highly Dynamic Spectrum

Routing with highly dynamic spectrum refers to the routing solutions that are designed for such CRNs where the holding time of the spectrum by CR users is much shorter than a communication flow duration. Spectrum availability changes frequently during the communication period. In such environments, routing techniques used in previous two subsections are not applicable. Thus new routing design is needed.

Opportunistic routing provides a promising paradigm for routing design in highly dynamic spectrum CRNs. As introduced in Section 2.3, opportunistic routing does not need to setup a fixed route which may introduce unaffordable route maintenance overhead when spectrum availability changes very fast. By taking the broadcast nature of wireless transmission into account, opportunistic routing dynamically chooses relay nodes to forward a packet. In such a manner, forwarding nodes can be selected according to their instantaneous spectrum conditions.

There have been some initial explorations on applying opportunistic routing to routing design in CRNs. In [48], the authors proposed a novel cost criterion to adapt the traditional opportunistic routing protocol ExOR [38] to CRNs. They identified that leveraging the expected transmission count (ETX) as the cost criterion to prioritize forwarding candidates ignores the opportunity to exploit potential CR links. The new cost is designed with the consideration of primary user activities. In their design, a packet is forwarded via both traditional links and CR links if the primary user is not present. A link cost is therefore defined as

$$c_{ij}(i) = \frac{T_{on_i} \cdot \frac{1}{p_{ij_{on}}} + T_{off_i} \cdot \frac{1}{p_{ij_{off}}}}{T} \tag{2.2}$$

where T_{on_i} is the time period when the CR link with a node i is not available and T_{off_i} is the time period when the CR link is valid. $p_{ij_{on}} = p_{ij}^{tr}$ is the delivery ratio using only a traditional link, while $p_{ij_{off}} = 1 - (1 - p_{ij}^{tr})(1 - p_{ij}^{cr})$ is the delivery ratio using both a traditional link and a CR link. Based on this link cost, the cost of a path from a specific node to the destination can be calculated with the sum of the costs of all the links on the path. Then the path cost of a node is used to rank the priority of this candidate node. The forwarding node selection can be performed dynamically according to the priority. The adapted opportunistic routing scheme performs well in CRNs. However, this scheme exploits CR links just as a supplement.

In [49], a spectrum aware opportunistic routing (SAOR) algorithm was proposed for CRNs with highly dynamic available links. A delay based node metric is designed to setup a prioritized relay candidate list in SAOR. The opportunistic nature of CR links is examined with respect to spectrum availability and a successful transmission rate. The opportunistic link is modelled as an M/Geo/1/∞/FCFS queue for deterministic packet sizes and M/M/1/∞/FCFS for variable packet sizes. With the analysis of the queue model, the statistical delay for an N-hop path with a variable packet size is derived as

$$OLT^{(N)} = \sum_{i=0}^{N} \frac{1}{\mu_i c_i - \lambda_i} \tag{2.3}$$

where λ_i, μ_i are the packet arrival rate and service rate for link i respectively. c_i is a product of spectrum availability and successful transmission rate for link i. Since for a CR node there are can be multiple paths with different lengths to the destination, the delay metric for the node is defined as

$$\min_{1 \le j \le T_i} \{OLT, (OLT_1^{(2)}, ..., OLT_{T_i^{(2)}}^{(2)}), ..., (OLT_1^{(N)}, ..., OLT_{T_i^{(N)}}^{(N)})\} \tag{2.4}$$

where $\{OLT_1^{(N)}, ..., OLT_{T_i^{(N)}}^{(N)}\}$ are delays for all possible N-hop paths. With such a node delay metric, the relay candidate list is setup. For the packet forwarding procedure, the random network coding method in MORE is exploited. SAOR considers a complete CRN environment with highly dynamic spectrum but only single channels are considered.

Multi-channel CRNs can provide more spectrum access opportunities, which is highly desirable in a highly dynamic spectrum environment. In Chapter 3, we present a multi-channel spectrum aware opportunistic routing (MSAOR) algorithm [85] in CRNs. Further the related batch transmission schemes are discussed in Chapter 4 and Chapter 5.

2.4 Multi-Channel Transmission

The use of multiple channels is often explored in traditional wireless networks [86, 87] to improve network performance, such as delay, throughput and transmission reliability. In CRNs, communications between CR users could fail since there is no guarantee on the availability of the spectrum for them. Moreover, due to channel fading and increased interference, data transmission in CRNs may also suffer packet loss and thus packet retransmission becomes more frequent than in other wireless networks. Multi-channel transmissions are especially useful for CRNs to obtain a higher spectrum access success probability and to fully explore access opportunities [88, 89]. In both of the two typical opportunistic routing protocols ExOR [38] and MORE [39], data packets are transmitted in batches for transmission coordination. Such batch transmission is also employed in some opportunistic routing schemes in CRNs. When opportunistic routing is designed for multi-channel CRNs, batch transmission schemes over unreliable links consisting of multiple channels should be carefully designed. In this section, we provide a review of some potential reliable multi-channel transmission schemes.

2.4.1 Multi-Channel ARQ

There are three basic automatic repeat request (ARQ) schemes [90] to provide reliable data transmission in wireless networks: stop-and-wait (SW-ARQ), go-back-N (GBN-ARQ), and selective-repeat (SR-ARQ). In SW-ARQ, the transmitter sends a packet to the receiver and waits the acknowledgment from the receiver. If the acknowledgment is negative (NACK), the transmitter will retransmit the packet. If the acknowledgment is a positive one (ACK), the transmitter will send the next packet. In GBN-ARQ, the transmitter continuously sends packets to the receiver. If a NACK is received, the negatively acknowledged packet and all the packets that have been sent after it need to be retransmitted. In SR-ARQ, the transmitter continuously sends packets and receives acknowledgments from the receiver. When a packet is negatively acknowledged, only that packet needs to be retransmitted. However, all these ARQ protocols are designed for single-channel communications.

To achieve reliable communications over unreliable channels in multi-channel scenarios, multi-channel ARQ protocols [91] have been proposed for packet transmission between a single-hop communication pair. These multi-channel ARQ protocols are designed based on the three basic ARQ strategies. Given M channels between the communication pair, the transmitter sends M packets every time, one on each channel. In multi-channel SW-ARQ, the transmitter waits for acknowledgments for the M packets after transmission. If a packet transmitted over channel k is negatively acknowledged, all the packets transmitted over channels k to M need to be retransmitted. In multi-channel GBN-ARQ, if a packet over channel k is negatively acknowledged, the packets transmitted over channels k to M in the same time slot and all the packets transmitted over the M channels after that slot are required to be retransmitted. In multi-channel SR-ARQ, only negatively acknowledged packets need to be retransmitted.

Ding et al. [92] studied multi-channel ARQ protocols with different packet-to-channel assignment rules and derived a general condition governing the packet-to-channel rule. In [93], Li et al. analyzed packet delay performance of multi-channel stop-and-wait ARQ through absorption analysis of transient Markov processes. They also demonstrated that a random scheduling strategy achieves better performance than a static scheduling strategy. Resequencing problems in multi-channel ARQ protocols were investigated in [94]. In [95], the authors proposed a multichannel fast ARQ scheme for transporting delay-sensitive flows in a multi-channel environment, which allows the retransmission only once but simultaneously using one or multiple channels. All these works provide deeper insight to the performance of multi-channel ARQ protocols.

However, when these multi-channel ARQ protocols are applied to batch transmission, the limited number of packets and efficient transmission over redundant channels should be carefully considered. Moreover, an ACK message is required for each packet in multi-channel ARQ transmission schemes. This results in the transmission under multi-channel ARQ schemes being highly dependent on feedback channels, which may be a scarce resource in CRNs.

2.4.2 Random Network Coding

Network coding schemes have been shown as an effective way to overcome unreliable links due to dynamic radio environments [39, 96]. In random network coding [97], the source node sends random combinations of original packets in a batch. The intermediate nodes re-combine the received coded packets and then send new coded packets. The destination recovers the original packets when it receives enough coded packets. For practical network coding, packet loss on network links and buffering in network codes are taken into account in random network coding [98, 99]

More specially, random network coding works as follows [100]. Assume there are K original data packets w_1, w_2, ..., w_K in the buffer of the source node. Each node performs a similar coding operation: the node stores the received packets in its buffer and coded packets are formed for transmission with random linear combinations of the stored packets in the buffer of the node whenever transmission occurs on an outgoing link. The coefficients of the linear combinations are uniformly selected from some finite field $\mathbb{F}_q$ at random.

As all coding operations are performed linearly, any packet u in the network can be written as a linear combination of the original packets w_1, w_2, ..., w_K, i.e. $u = \sum_{k=1}^{K} \gamma_k w_k$. Vector $\gamma = (\gamma_1, \gamma_2, ..., \gamma_k)$ is termed as the *global encoding vector* of packet u. This global encoding vector is sent with packet u as side in formation in the packet header. Such introduced overhead is negligible if the packet size is sufficiently large. For simplicity, nodes are usually assumed to have infinite buffers, but this can be modified to that a node saves the newly received packet only if the global encoding vector of this packet is linearly independent with the global encoding vectors of the packets already stored in its buffer. With such a modification, the buffer size required is only K packets.

The destination node collects packets and recovers the original data packet when it has got K packets with linearly independent global encoding vectors. The decoding can be performed by via Gaussian elimination. The random network coding procedure is summarized in Figure 2.9 [100].

For example, supposing there are three ($K = 3$) original packets w_1, w_2, w_3, one of the coded packets at the source can be $u = 2w_1 + w_2 + 3w_3$, whose global encoding vector at this stage, before it may be further re-encoded during transmission in the network,

Initialization:

- The source node stores the original packets $w_1, w_2, \dots, w_K$ in its buffer.

Operation:

- When a packet is received by a node,
  - the node stores the packet in its buffer.
- When a packet transmission occurs on an outgoing link of a node
  - the node forms the coded packet from a random linear combination of the packets in its buffer. Suppose the node has L packets $u_1, u_2, \dots, u_L$ in its buffer. Then the packet formed is

$$u_0 := \sum_{l=1}^{L} \alpha_l u_l ,$$

    where α_l is chosen according to a uniform distribution over the elements of $\mathbb{F}_q$. The packet's global encoding vector γ, which satisfies $u_0 = \sum_{k=1}^{K} \gamma_k w_k$, is placed in its header.

Decoding:

- The destination node performs Gaussian elimination on the set of global encoding vectors from the packets in its buffer. If it is able to find an inverse, it applies the inverse to the packets to obtain $w_1, w_2, \dots, w_K$; otherwise, a decoding error occurs.

Figure 2.9: Random linear network coding procedure

is (2,1,3). Three independent packets received at the destination node are, for instance, $u_1 = w_1 + w_2 + 2w_3$, $u_2 = 2w_1 + w_2 + 2w_3$, $u_3 = 3w_1 + 4w_2 + 5w_3$, whose global encoding vectors are (1,1,2), (2,1,2) and (3,4,5) respectively. The destination node can recover the original data packets as

$$\begin{pmatrix} w_1 \\ w_2 \\ w_3 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 2 \\ 2 & 1 & 1 \\ 3 & 4 & 5 \end{pmatrix}^{-1} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} = \begin{pmatrix} -1 & 1 & 0 \\ -\frac{4}{3} & -\frac{1}{3} & \frac{2}{3} \\ \frac{5}{3} & -\frac{1}{3} & -\frac{1}{3} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} = \begin{pmatrix} -u_1 + u_2 \\ -\frac{4}{3}u_1 - \frac{1}{3}u_2 + \frac{2}{3}u_3 \\ \frac{5}{3}u_1 - \frac{1}{3}u_2 - \frac{1}{3}u_3 \end{pmatrix} \tag{2.5}$$

Such random coding can blur packet information, making packets transmitted over networks equal from the perspective of packet information. This means that there will be no requirements on which particular packets are received by the receiver, but only on receiving enough independent coded packets. An alternative way with a similar function is to use fountain codes in 1-hop transmission as in [101] but fountain codes do not extend readily to a network setting [102].

In Chapter 4 we focus on multi-channel batch transmission schemes in a single hop CRN. Then the work is extended to a two hop CRN in Chapter 5, combined with different routing strategies.

2.5 Summary

In this chapter, we discussed the background and reviewed related works on CR routing design. Although there have been some works on CR routing design, there are still some important issues that need to be further studied. While most of existing CR routing protocols assume that the variation of spectrum availability is relatively slow, there is little work on routing design in highly dynamic spectrum CRNs. Leveraging opportunistic routing technique, we will design a routing algorithm for multi-channel CRNs with fast varying spectrum availability in Chapter 3. As batch transmission is a key component of some opportunistic routing algorithms, the efficiency and performance of batch transmission schemes have important impacts on routing. In Chapter 4 and Chapter 5, we will conduct comprehensive discussion and analysis of efficient and reliable multi-channel batch transmission schemes in CRNs under lossy environment. In CRNs, CR users usually need to switch among different available channels, which introduces switching delay. Although some delay metric based CR routing algorithms take such channel switching into account, these algorithms require all channel information at each node along a route path to be disseminated to the destination for delay metric calculation. We presents a maximum flow-segment based channel assignment in Chapter 6, which achieves minimized channel switches with efficient channel information dissemination.

Chapter 3

Opportunistic Routing in Multi-Channel Cognitive Radio Networks

3.1 Overview

Cognitive radio is an exciting technology to improve spectrum utilization in wireless communications. In cognitive radio networks, routing protocol design, as an essential networking technology, plays an important role to improve end-to-end performance over multiple hops. A unique challenge for routing design in CRNs is the dynamics of channel availability. Most of the routing algorithms for CRNs are designed jointly with channel assignment. A general assumption in the design of these routing algorithms is that spectrum availability varies slowly in CRNs. However, in practice spectrum availability can change very fast due to primary users' activities. Such fast variations make most of existing CR routing protocols fail. One class of routing paradigm, the opportunistic routing [38, 39], is able to achieve low delay and high reliability by taking advantage of the broadcast nature of radio transmission. Opportunistic routing provides a promising routing paradigm for CRNs with fast variation of spectrum availability.

In this chapter, we propose a multi-channel spectrum aware opportunistic routing (MSAOR) algorithm for CRNs. The proposed MSAOR algorithm attempts to broadcast packets on every available channel of the CR link, exploiting the benefits of multiple channels, as well as radio broadcasting nature. This can reduce the link delay and further reduce the end-to-end delay. MSAOR also achieves higher throughput than SAOR [49]. In our analysis, the opportunistic link consisting of multiple channels is modeled as an M/Geo/1 queue. In order to model practical multi-user networks, channel access probability based on network contention level is introduced to characterize the opportunistic nature of the CR link, in addition to channel availability and successful transmission rate. In our link delay analysis, the access delay is integrated into queuing delay. Our analysis shows that the delay of the link with multiple channels is lower than that with only one single channel. Based on the link queueing model, the node delay metric is derived and used for forwarder candidate list setup. Additionally, we exploit network coding in our MSAOR algorithm to further enhance the throughput.

Simulation results show that 1) the end-to-end delay performance of MSAOR is better than that of SAOR, by up to 50%; 2) The maximum supported offered load is increased by up to 190%; 3) MSAOR is more robust than SAOR.

The remainder of this chapter is organized as follows. Section 3.2 introduces the system model with the link model and traffic model. We discuss and analyze the delay based node metric and describe the details of MSAOR algorithm in Section 3.3. In Section 3.4, the simulation results are presented and discussed. Finally, we summarize our work in Section 3.5.

3.2 System Model

We consider a multi-hop multi-channel multi-radio cognitive radio network. Multi-radio CR nodes in the network can work on multiple channels simultaneously.

In the following parts of this section, we first describe the link model to capture the opportunistic nature of CR links, which is based on three metrics associated with the process of packet transmission over CR links. Then we discuss the traffic model based on queueing theory.

3.2.1 Link Model

The opportunistic nature of the links that connect CR users includes three major metrics: channel availability, channel access successful probability and successful transmission rate. In a multi-channel CRN, these three metrics can be different with different channels. When modelling the opportunistic link, the following should be therefore considered:

1) Channel Availability

In CRNs, the availability of a channel depends on primary users' activities. The CR users can only use the channels currently unoccupied by the primary users and have to quit the channels if primary users need to use the channels again. We assume that there are totally K channels in the system, which are potentially available for the link between two CR nodes in a CRN. A CR node is assumed to have adequate spectrum sensing capability that can provide the network layer with the available channel information. The probability that a channel, say k, is available for CR users is denoted as θ_k

The primary users' activity can be modelled as an embedded two-state discrete-time Markov chain and thus we can have the Markov channel model as in Figure 3.1 [103] for the availability of a potential channel in a CRN. In the figure, the state "busy" means that the channel is being used by primary users currently and thus it is not available for

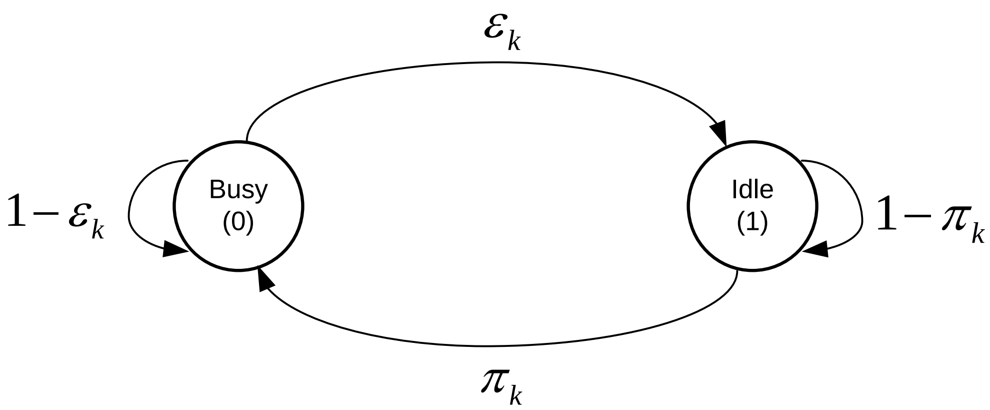


Figure 3.1: The Markov channel model

CR users. The state "idle" means that the channel is not being used by primary users currently and thus it is available for the communication between CR users. A channel, say k, transits from state "busy" to state "idle" with probability ε_k, and keeps the state "busy" unchanged with probability $1-\varepsilon_k$; the channel transits from state "idle" to state "busy" with probability π_k, and keeps the state "idle" unchanged with probability $1-\pi_k$.

Therefore based on the above Markov channel model, we can obtain the probability θ_k that channel k is available as

$$\theta_k = \frac{\varepsilon_k}{\varepsilon_k + \pi_k} \tag{3.1}$$

2) Channel Access Successful Probability

For channel k, it can be accessed with probability ω_k, due to the contention from other neighboring multiple users.

When a node wants to send a packet on channel k, it must contend the channel with all of its neighbors within one hop who also have packets to send on the same channel. The channel contention level depends on both the number of neighbors and the traffic of these neighboring nodes.

Note that for channel access considered here is in a general contention based manner. The collision of packets is not considered in the analysis here, which can be reduced or avoided by some mechanisms such as backoff. This can be different with the slotted ALOHA. However, we do not deal with the details of contention mechanisms here.

We assume that there are totally N nodes in the one-hop neighborhood of node A, $node_i \in Nr(A)$, where $Nr(A)$ denotes the set of node As one-hop neighbors. Further, the probability that node i has packets to send on channel k is denoted as $\alpha_{k,i}$. Given that channel k is available for node A, the probability that node A can access the channel successfully, is

$$\omega_k = \frac{1}{1 + \sum\limits_{node_i \in Nr(A)} \theta_{k,i}\alpha_{k,i}} \tag{3.2}$$

where $\theta_{k,i}$ is the probability that channel k is available for node i (Different nodes can have different channel availability because they may influenced by different primary users or different distances to one same primary user), while $\sum\limits_{node_i \in Nr(A)} \theta_{k,i}\alpha_{k,i}$ is the expected number of nodes that would contend for channel k with node A. In practice, if there is no information about traffic on channel k for node i, $\alpha_{k,i}$, channel access successful probability of channel k can be approximately estimated with the number of one-hop neighbors by

$$\omega_k \approx \frac{1}{1 + \sum\limits_{node_i \in Nr(A)} \theta_{k,i}} \tag{3.3}$$

3) Successful Transmission Rate

Wireless transmission could suffer transmission failure due to wireless channel fading. Different channels may experience different fading. If the SNR at the receiver is lower than the threshold required, the transmission will fail.

The successful transmission rate over channel k is defined as

$$\begin{aligned} \delta_k &= \Pr\{SNR_r^{ch_k} \geq \kappa\} \\ &= \Pr\{\frac{|h_k|^2 (\frac{d}{d_0})^{-\alpha} P_s}{N_0} \geq \kappa\} \end{aligned} \tag{3.4}$$

where κ is the required threshold of SNR. The absolute value $|h_k|$ denotes the channel gain over channel k, which is influenced by the fading effect. Moreover, d is the distance between transmitter and receiver, d_0 is a reference distance for antenna far field, α is the path-loss exponent, P_s is transmitted signal power, and N_0 is determined by the power spectral density of the AWGN noise.

By now, we have introduced channel availability, channel access successful probability and successful transmission rate as three metrics of CR links. Next we will discuss the arrival and departure process of the traffic.

3.2.2 Traffic Model

We assume that the network works in a slotted mode. The size of packets transmitted between CR users is deterministic. The arrival traffic is a Poisson process with arrival rate λ. For an ideal link, it can be modeled as an M/D/1 queue. However, considering the opportunistic nature of the link in CRNs, the following process should be considered.

When a node has a packet to send, it attempts to send the packet on all the channels. The flow chart of the packet transmission process over a CR opportunistic link on channel k is shown in Figure 3.2.

First, the node checks whether channel k is available via performing spectrum sensing. If the channel is not available, the node waits in this time slot and sense the channel in the next time slot. Such a process is performed by the node in each time slot until the channel is available. In each time slot, the channel availability checking result for channel k is a Bernoulli trial, which is associated with a success probability θ_k. The number of time slots, denoted as X, the node needs to wait before the channel is available, follows a geometric distribution, $Geo(\theta_k)$.

$$X \sim Geo(\theta_k) \tag{3.5}$$

Then, if a channel is available, the node tries to access the channel in the second step. The node needs to contend the access opportunity to this channel with other neighboring nodes who also wants to transmit packets on this channel. The contention

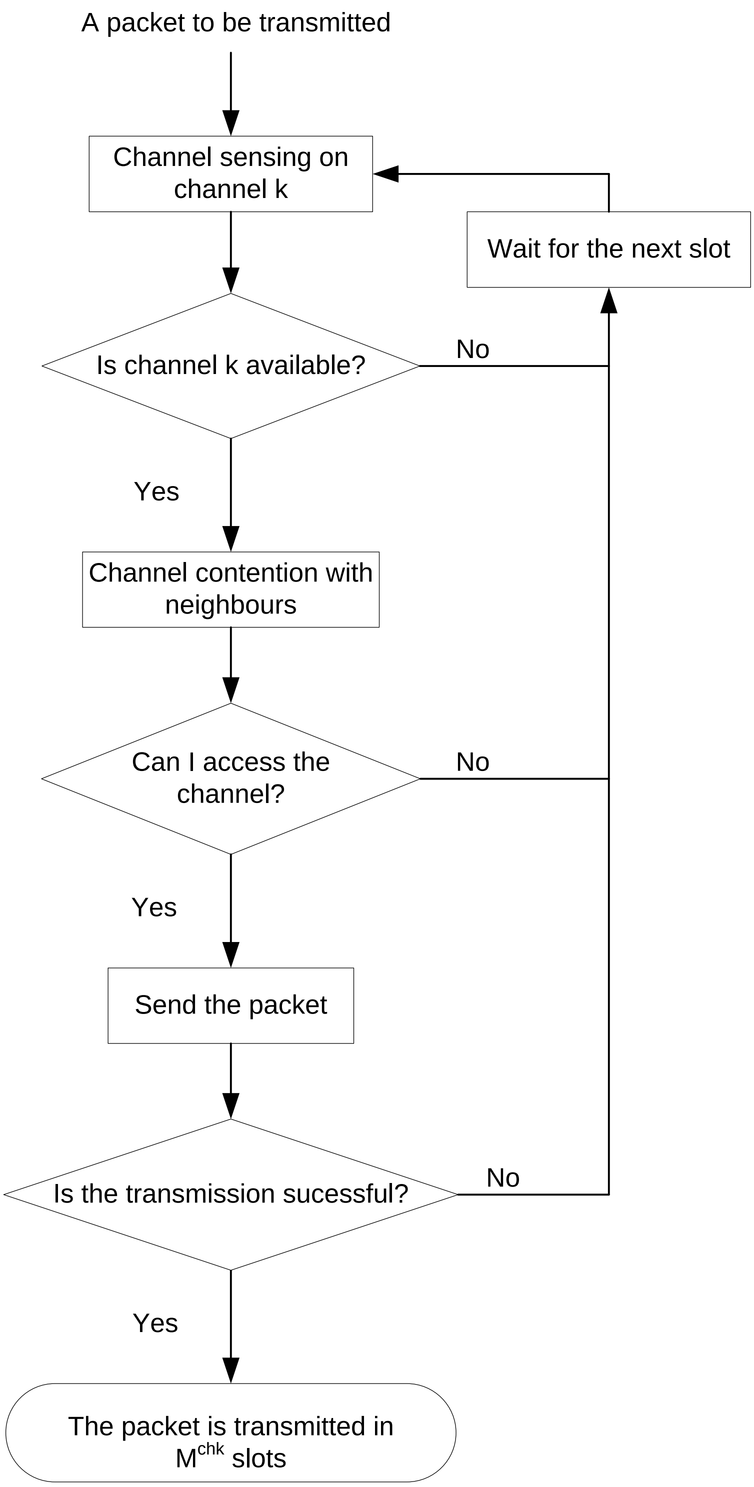


Figure 3.2: Packet transmission process: the figure shows a flow chart of how a packet is transmitted over a CR opportunistic link on channel k

is with successful access probability ω_k for this node as determined in equation (3.2) or (3.3) with different information. The number of time slots that the node needs to wait for successful access to the channel is denoted as Y, which follows a geometric distribution $Geo(\omega_k)$. This is due to the fact that each access attempt is a Bernoulli trial with successful probability ω_k.

$$Y \sim Geo(\omega_k) \tag{3.6}$$

The number of time slots that the node needs to wait to successfully access channel k at this stage, combined with channel availability in the first step, is the summation from X_1 to X_Y.

$$\sum_{i=1}^{Y} X_i \sim Geo(\theta_k \omega_k) \tag{3.7}$$

Finally, if the node successfully accesses channel k, it sends the packet over the channel. The transmitted packet is with a loss probability of $(1 - \delta_k)$. If the packet is lost due to channel fading, it needs to be retransmitted. The number of time slots with retransmission to get the packet transmitted successfully is Z. This is still a Bernoulli trial for each transmission with a success probability δ_k. As a result, Z follows a geometric distribution, $Geo(\delta_k)$. The number of time slots for the node to successfully transmit one packet, combined with the previous two steps, is given by the summation of X_{1,Y_1},...,X_{Y_1,Y_1}, X_{1,Y_2},...,X_{Y_2,Y_2}, ..., X_{1,Y_Z},...,X_{Y_Z,Y_Z}.

$$M^{ch_k} = \sum_{j=1}^{Z} \sum_{i=1}^{Y_j} X_{i,Y_j} \sim Geo(\theta_k \omega_k \delta_k) \tag{3.8}$$

Note that we can understand the overall process in the following way. A packet is sent successfully only when all of the above three steps are successful. The probability that a node finds the channel is available, it can access the channel and it send the packet successfully, is $p_k = \theta_k \omega_k \delta_k$. Thus, when a node tries to send a packet, it is a Bernoulli trial with successful rate p_k . The number of slots for the node to send the packet follows a geometric distribution with parameter $\theta_k \omega_k \delta_k$.

The packet is attempted to be sent on every channel and the number of waiting time slots on channel k is M^{ch_k} as above. Therefore the number of waiting slots for transmitting a packet over the CR link, M, is the minimum one among the delays over all the channels.

$$M = \min_k M^{ch_k} \sim Geo(p) \tag{3.9}$$

where

$$p = 1 - (1 - p_1)(1 - p_2)...(1 - p_K) = 1 - \prod_{k=1}^{K} (1 - p_k) \tag{3.10}$$

One way to understand the above conclusion is as follows. All the channels combined together can be seen as a virtual channel between the transmitter-receiver pair. A packet fails to be sent over the virtual channel only when it fails on all the K channels with the probability $p_{fail} = (1 - p_1)(1 - p_2)...(1 - p_K)$. So the successful probability for a packet sent over the virtual channel is $p = 1 - p_{fail}$. A packet being sent over the K channels is a Bernoulli trial with successful rate p and the number of slots for the node to send the packets follows a geometric distribution with parameter p.

Therefore, considering the arrival traffic is a Poisson process, the queue model for the opportunistic link in a CRN is an M/Geo/1 queue with $Geo(p)$.

3.3 Multi-Channel Spectrum Aware Opportunistic Routing (MSAOR) Algorithm

In the previous section, we analyzed the opportunistic nature of CR links and setup an M/Geo/1 queue model for the traffic arrival and service process. Now, we present our proposed multi-channel spectrum aware opportunistic routing algorithm based on the analysis of the queue model.

The main tasks of opportunistic routing design are 1) setup a prioritized forwarder list and 2) forwarding procedure design. In the first subsection we derive the route metric for forwarder selection, while in the second subsection we provide the algorithm for packet forwarding. Most of that for the work in this part is similar to SAOR but with multi-channel support extensions.

3.3.1 Opportunistic Link Transmission Metric over Multiple Channels (MC-OLT)

The service time of the direct opportunistic link between two CR nodes is denoted as S. Given that the packet size is deterministic and the system works in a slotted mode with slot duration Δt, we have $S = M\Delta t$.

The derivation of MC-OLT is similar to OLT in [49] as below.

The average service time is

$$\begin{aligned} E[S] &= E[M\Delta t] \\ &= \Delta t \cdot E[M] \\ &= \frac{\Delta t}{1 - \prod_{k=1}^{K} (1 - p_k)} \\ &= \frac{\Delta t}{p} \end{aligned} \tag{3.11}$$

The second moment of service time S is

$$\begin{aligned} E[S^2] &= E[(M\Delta t)^2] \\ &= \Delta t^2 \cdot E[M^2] \\ &= \Delta t^2 (Var[M] + E[M]^2) \\ &= \Delta t^2 (\frac{1-p}{p^2} + \frac{1}{p^2}) \\ &= \frac{\Delta t^2 (2-p)}{p^2} \end{aligned} \tag{3.12}$$

The waiting time in the queue thus can be calculated with *Pollaczek-Khinchin (P-K) formula* [90],

$$W_q = \frac{\lambda \Delta t^2 (2-p)}{2p(p-\lambda\Delta t)} \tag{3.13}$$

The total waiting time, in queue and in service, is

$$\begin{aligned} MC-OLT &= E[S] + W_q \\ &= \frac{\Delta t}{p} + \frac{\lambda \Delta t^2 (2-p)}{2p(p-\lambda\Delta t)} \\ &= \frac{\Delta t(2-\lambda\Delta t)}{2(p-\lambda\Delta t)} \end{aligned} \tag{3.14}$$

Further, $MC-OLT^{(2)}$ and $MC-OLT^{(N)}$, which are the delay metrics of 2-hop path and N-hop path, are in the similar form of $OLT^{(2)}$ and $OLT^{(N)}$ in [49], respectively.

$$MC-OLT^{(2)} = (E[S_1] + \frac{\lambda E[S_1^2]}{2(1-\lambda E[S_1])}) + (1 + \frac{E[S_2]-1}{1-\lambda E[S_2]}) \tag{3.15}$$

$$MC-OLT^{(N)} = (E[S_1] + \frac{\lambda E[S_1^2]}{2(1-\lambda E[S_1])}) + \sum_{i=0}^{N-2} (1 + \frac{E[S_{i+2}]-1}{1-\lambda E[S_{i+2}]}) \tag{3.16}$$

where $E[S_i]$ is the mean service time of the i^{th} hop.

Finally, with $MC-OLT$, $MC-OLT^{(2)}$,..., and $MC-OLT^{(N)}$, we get the node delay metric

$$\begin{aligned} \min_{1\le j\le T_i} \{&MC-OLT, (MC-OLT_1^{(2)}, ..., MC-OLT_{T_i^{(2)}}^{(2)}), \\ &..., (MC-OLT_1^{(N)}, ..., MC-OLT_{T_i^{(N)}}^{(N)})\} \end{aligned} \tag{3.17}$$

where $MC-OLT_{T_i^{(N)}}^{(N)}$ denotes the total waiting time for the packet of the i^{th} node transmitted over $(T_i^{(N)})^{th}$ N-hop path in multi-channel CRNs.

3.3.2 MSAOR Algorithm

The MSAOR algorithm is an extension of SAOR for multi-channel CRNs. It also exploits the idea of random coding and the packet forwarding mechanism in MORE [39] to avoid packet duplicates. The MSAOR algorithm is listed as *Algorithm 3.1* in Figure 3.3.

Algorithm 3.1 MSAOR

```
Source collects Θ, Ω and Δ
Source makes candidate list in NL
  for every batch of packets do
    if destination doesn't received enough innovative packets
      source node code packets and broadcast the coded packet over multiple channels
      for every received packet relay node z do
        if relay node z is in NL
          if the received packet is innovative
            relay node z saves the packet in buffer
              relay node z codes packets in the buffer into a new packet
              relay node z broadcasts the new packet over multiple channels
        else discard the received packet
      for every received packet destination do
        if the received packet is innovative
          save the packet in buffer
        else discard the received packet
    else
      destination decode buffered packets
      break
Destination sends ACK and Source moves to next batch
```

Figure 3.3: Algorithm 3.1 MSAOR

In the multi-channel CRN with (n+1) CR nodes and K channels, the spectrum map is a multi-layer map, corresponding to the single-layer spectrum in a single-channel CRN. The symbol $\Theta = [\theta_{ij}^k]$ is a (n+1) by (n+1) by K matrix. The symbol θ_{ij}^k denotes the available probability of channel k for the opportunistic link between node i and node j.

The symbol $\Omega = [\omega_i^k]$ is a (n+1) by K matrix. The symbol ω_i^k denotes for the successful probability for node i to access channel k. The symbol $\Delta = [\delta_{ij}^k]$ denotes the successful transmission rate information between node i and node j.

The above link information is collected by the source node at the beginning and it can be updated when it is necessary. With this information, the source node calculates the node delay metric according to equation (3.17). Then the forwarder list is setup based on the node delay metric; for example, only nodes with smaller delays can forward the packet. Packets arriving at the source are sent in a batch with random network coding. Information such as the forwarder list, batch ID, flow ID and code vector will be added in the packet head. The coded packet will then be attempted to broadcast on every channel until it is sent out.

When a relay node receives a packet, it first checks if it is in the forwarder list and if the packet is innovative. An innovative packet means that the code vector in the head of this packet is linearly independent with those of packets in the buffer of this node. If the packet is an innovative one, the relay node buffers the packet, then generates and broadcasts a coded packet.

When the destination receives enough innovative packets, which means the number of such packets equals to the number of packets in a batch, it decodes the packets and sends the ACK for this batch. Once the batch is acknowledged, all the transmissions for packets in this batch will stop and the nodes move to transmission for the next batch.

3.4 Simulation and Evaluation

To compare our proposed algorithm with the existing SAOR, we setup the simulation in similar environment and parameters to that in [49], but with some multiple channel extension.

We evaluate the MSAOR algorithm in a three-node network (source node S, relay node R and destination node D) similarly to SAOR [49] but with multiple potential channels. Each CR user is equipped with the same number of transceivers as the number of potential channels and thus a CR user can transmit packets on multiple channels concurrently.

The distance between the source node and the destination node is twice of that between the relay node and the other two (i.e., $d_{SD} = 2d_{SR} = 2d_{RD}$).

CR users work in a slotted system and the packet size is deterministic. The arrival traffic at the source node is a Poisson process. For each channel, the ideal link service rate is 500 packets per second. Here the ideal link service rate is the service rate when the channel is available, a CR user accesses the channel successfully and no packets are lost during data transmission .

Each channel can be accessed with a probability 0.5 [49] when it is unoccupied by primary users. Successful transmission rate is 0.8 for the link between the source node and the relay node and the link between the relay node and the destination. The direct link for the source node to the destination has a successful transmission rate 0.3.

We simulate the algorithm with Matlab programming. In the simulation, we simulate packet encoding and decoding based on coding vector associated each packet. Channel availability, channel access and transmission success are controlled by different random variables.

We follow the simulation of SAOR and do not make statistical errors analysis. But some basic insights can be obtained from the simulation results of OR_SM in Chapter 5.

3.4.1 Maximum Supported Offered Load

Figure 3.4 shows the end to end delay performance of the CRN with different numbers of channels. When the number of channels is one, it corresponds to SAOR. Given the number of channels in a CRN, there is a maximum threshold for the packet arrival rate (offered load). If the arrival rate exceeds the threshold, the delay will increase sharply with the arrival rate. We refer to the maximum threshold for the packet arrival rate as the *maximum supported offered load*. The maximum supported offered load indicates that: the packet transmission in the CRN can be conducted with a reasonable and acceptable delay if the packet arrival rate is lower than it; but there will be no delay performance guarantee for any transmission with a higher packet arrival rate. Thus the maximum supported offered load can serve as a measure for the achievable throughput of a CRN under some given delay performance constraint.

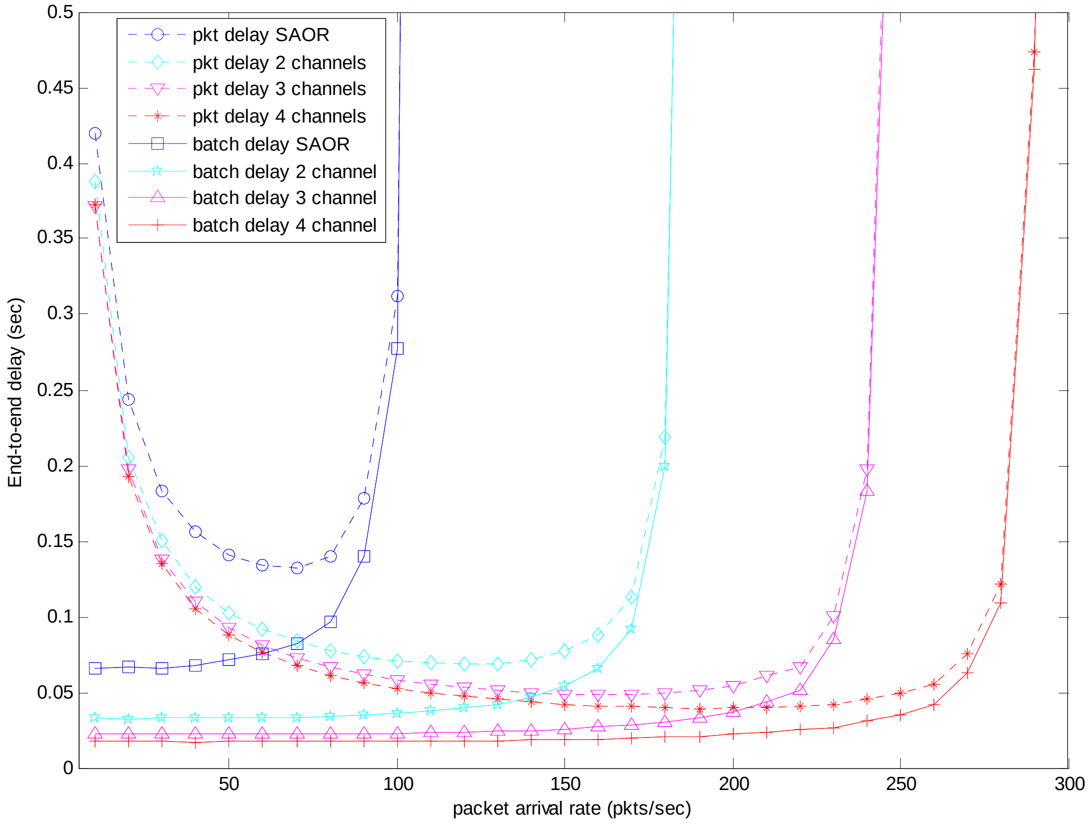


Figure 3.4: End-to-end delay as a function of packet arrival rate

It is shown in Figure 3.4 that the maximum packet arrival rate, i.e. the maximum supported offered load, for the single channel opportunistic routing algorithm, SAOR, is approximately 100 packets per second. For MSAOR with 2, 3 and 4 channels, the maximum supported offered loads are about 180, 240 and 290 packets per second, respectively. Compared with SAOR, the maximum supported offered load associated with MSAOR is improved by up to 190%. This means MSAOR is more suitable to work in the CRNs with heavier traffic than SAOR, by exploiting the benefits of multiple channels.

Furthermore, take the maximum supported offered load of SAOR as the benchmark. When 2 channels are exploited with MSAOR, the one extra channel help to improve the maximum supported offered load from 100 packets per second to 180 packets per second, by 80%. When 3 channels are exploited, the second extra channel provides a further improvement of the maximum supported offered load, about 60%.

Finally, when 4 channels are used for the transmission in the CRN, the third extra channel just bring a 50% improvement of the maximum supported offered load. It can be found that the improvement of the maximum supported offered load introduced by providing one extra channel become smaller as the total number of channels in the network increases. This indicates that there is a tradeoff between the improvement of the maximum supported offered load and the improvement of the efficiency of added extra channels (i.e. spectrum resource): more channels can provide higher maximum supported offered load, but result in lower efficiency of each channel.

3.4.2 End-to-End Packet Delay Performance

In Figure 3.4, the four dashed curves show the end-to-end packet delay performance of SAOR and MSAOR with 2, 3 and 4 channels. It can be seen that while the number of channels increases, the packet delay becomes lower. The packet delay performance gaps between different schemes become larger and larger as the packet arrival rate increases.

Examining the minimum delays that different schemes can provide, we can see that SAOR can achieve the minimum delay of about 0.14 second; MSAOR with 2, 3 and

4 channels enjoys the minimum delays for approximately 0.07, 0.05 and 0.04 second. The minimum delay is reduced by 50%, 64% and 71%. Moreover, MSAOR achieves the minimum delays when packet arrival rates are 130, 170 and 210 packets per seconds, while SAOR get the minimum delay at 70 packets per second for packet arrival rate. In a word, MSAOR can achieve lower minimum delays at higher packet arrival rates, compared to SAOR.

Even in the range of packet arrival rate lower than both the maximum supported offered loads of SAOR and MSAOR, MSAOR suffers a much lower packet delay compared to SAOR. If we take the arrival packet rate, 70 packets per second, at which the minimum delay of SAOR is achieved, it can be found that MSAOR reduces the packet delay by about 50%. If we look at the packet delay when the packet arrival rate is 90 packets per second, after which there is a sharp jump of the packet delay of SAOR, MSAOR can provide a reduction of about 65%.

It can be observed in Figure 3.4 that given the number of channels, the end-to-end packet delay falls down at first and then increases as the packet arrival rate increases. The reason behind this is that batching time is considered when calculating the packet delay. In both SAOR and MSAOR, packet transmission is batch based. Packets at the source node will not be sent until all the packets in a batch arrive. Thus there is a batching time for a packet to wait the arrival of other packets in the same batch. The packet delay is the time interval between the time the packet arrives and the time the batch including the packet is acknowledged. When the packet arrival rate increases, the batching time becomes smaller. From Figure 3.4 we can find that the packet delays of both SAOR and MSAOR experience a quick drop when the packet arrival rates are lower than 50 packets per second, and then decrease slowly before arrive the minimum values. This discloses that before packet arrival rates exceed 50 packets per second, the batching time account for a major part of packet delay. Since the numbers of packets in a batch are same, both SAOR and MSAOR suffer the same batching time and have the same critical packet arrival rate at 50 packets per second.

3.4.3 End-to-End Batch Delay Performance

In Figure 3.4, batch delay performance with different numbers of channels is also provided. The batch delay is defined as the time interval between the time the last packet of the batch arrives and the time the packet is acknowledged.

In Figure 3.4, it can be seen that the batch delay increases slowly with the packet arrival rate when the packet arrival rate is lower than some threshold. The varying trend of the curves is similar to the network in which packets are not batched. This means the batch is delivered like a big virtual packet. The batch delay is lower than the packet delay as shown in the figure.

Moreover, the gap between the batch delay and packet delay is reduced with the increase of the packet arrival rate. When the packet arrival rate is large enough, greater than the threshold, batch delay and packet delay are almost the same. This is due to the fact that, with the packet arrival rate increasing, there are more packets queueing in the buffer to be sent. Thus when a batch is acknowledged the next batch needs less or no time to wait for the arrival of all the packets in it.

3.4.4 Delay Performance with Different Channel Availability

In the next experiment, we evaluate the end-to-end packet delay for both SAOR and the MSAOR with different channel availability. We provide 4 channels when evaluating MSAOR.

The results in Figure 3.5 show that the packet delay increases as the channel availability becomes smaller, for both algorithms. Given the delay requirements, our proposed MSAOR is more robust to the variation of channel availability. For example, provided

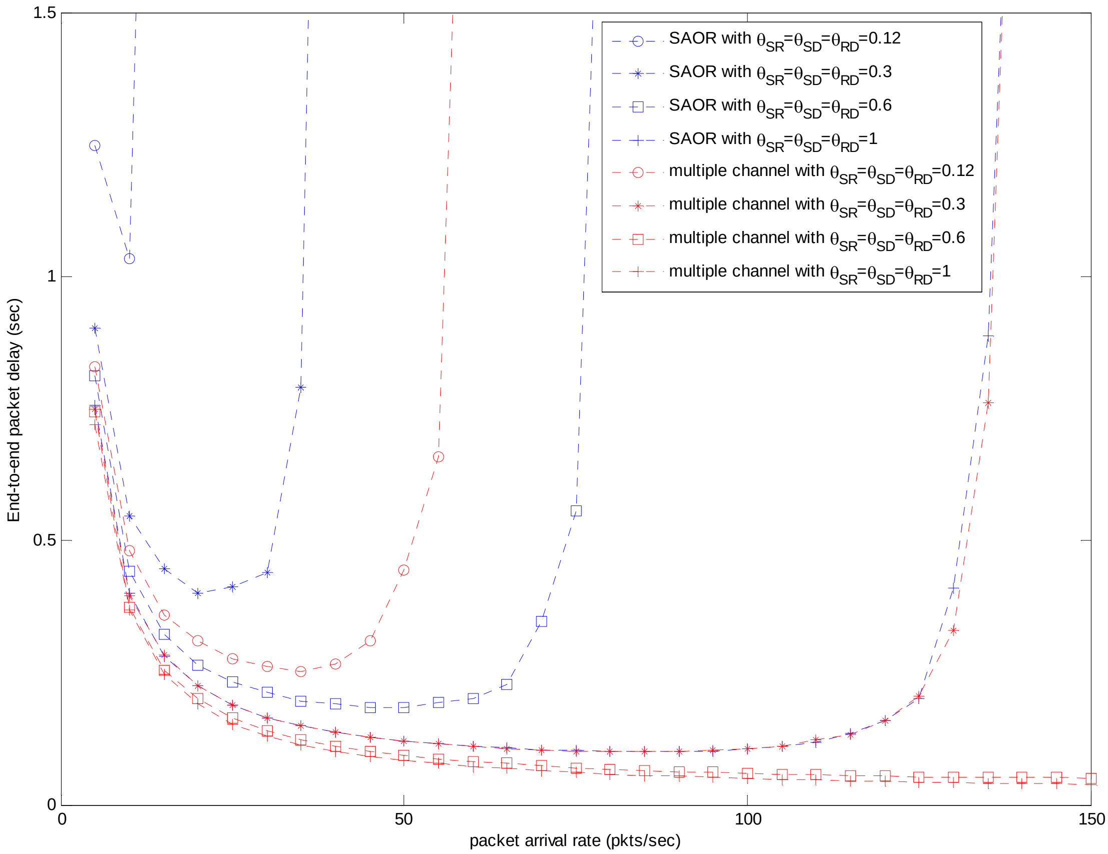


Figure 3.5: End-to-end packet delay with different channel availability

the delay requirement is 0.2 second with 50 packets per second, channel availability must be greater than 0.6 with SAOR. Working with MSAOR, even if the channel availability is only 0.3, the delay is still much lower than the requirement. Also it can be seen in Figure 3.5 that MSAOR with channel availability 0.3 performs almost as well as SAOR with channel availability 1. The results reveal that by exploiting multiple channels, MSAOR can compensate the lost performance due to lower channel availability.

3.5 Summary

In this chapter, we proposed the MSAOR algorithm for multi-channel CRNs. Three metrics are identified to capture the opportunistic nature of a CR link. The process of packet arrival and service was modeled as an M/Geo/1 queue. Taking advantage of multiple channels, the link delay is lower than that associated SAOR according to the analysis of service time based on the M/Geo/1 queue model. Simulation results verified that MSAOR delivered lower end-to-end delay and was able to support heavier traffic. We also demonstrated the algorithm can compensate for lost performance due to lower channel availability, by exploiting multiple channels.

In our proposed MSAOR algorithm, packets are transmitted in batches. Such batch based transmission is a general key component in some opportunistic routing algorithms. The performance of batch transmission scheme has a significant impact on the performance of these routing algorithms. In MSAOR, all idle channels are exploited for packet transmission rather than that only one channel is used in some existing transmission schemes, which is important to accelerate batch transmission in CRNs. In the next two chapters, we will focus on more efficient batch transmission scheme design and analysis.

Chapter 4

Efficient Data Transmission with Random Linear Coding in Multi-Channel Cognitive Radio Networks

4.1 Overview

Efficient data transmission in cognitive radio network is critical for cognitive radio users to communicate with each other in an opportunistic manner. Even with successful access to required channels, the transmission could still suffer from failures due to channel fading. In the previous chapter, we presented an opportunistic routing algorithm for multi-hop data transmission in a multi-channel CRN. We also conducted the analysis of link based packet delays using an M/Geo/1 model. However, packets in the proposed algorithm in Chapter 3, as well as in some other opportunistic routing algorithms, are transmitted in batches. Hence batch transmission design and batch based performance analysis, such as batch transmission delay, can provide a deeper insight to those routing algorithms than the link based packet delay analysis.

In this chapter, we consider practical fading channels in a cognitive radio environment, where data transmissions of CR users may experience packet loss due to bit errors, and channel availability varies with time depending on primary users' activities. We introduce random linear coding to multi-channel batch transmission between single-hop communication pairs over lossy channels in CRNs. We design a coded scheme, which in essence can be considered as an alternative to retransmission. This scheme is referred to as the *random linear coding based multi-channel transmission* scheme (*RLC-MCT*). In RLC-MCT the source node sends random linear combinations of original packets in a batch, similar to the way it is done in random network coding. RLC-MCT, therefore, can be easily extended to network coding in multi-hop networks.

We also provide two additional multi-channel batch transmission schemes, exploiting multi-channel ARQ. They are referred to as the *multi-channel ARQ based with single-copy retransmission of lost packets* scheme (*SC-MARQ*), and the *multi-channel ARQ based with multi-copy retransmission of lost packets* scheme (*MC-MARQ*). We analyze and derive batch delay performance associated with these three schemes. Analytical results, validated by simulations, show that the proposed RLC-MCT outperforms SC-MARQ and MC-MARQ. Moreover, RLC-MCT requires only one ACK for all packets in a batch, while SC-MARQ and MC-MARQ require an ACK for each packet. This makes RLC-MCT less dependent on feedback channels.

The remainder of this chapter is organized as follows. In Section 4.2, the system model and different batch transmission schemes are introduced. Then we analyze batch delay associated with the three transmission schemes in Section 4.3. In Section 4.4 performance evaluation is conducted. The work in this chapter is summarized in the last section.

4.2 System Model and Multi-Channel Transmission Schemes

4.2.1 System Model

We consider source node S and destination node D in a CRN, communicating over a single-hop multi-channel wireless link as in Figure 4.1. Node S has N packets (one batch) in its buffer ready to send to node D. There are totally K potential channels in the channel pool $CP = \{ch_i | i = 1, 2, ..., K\}$ for the communications between node S and node D. Some, even all, of these potential channels however could be unavailable due to primary users' activities. We assume that the probability that ch_i is available is θ_i and in each slot there are k ($k \in \{0, 1, 2, ..., K\}$) available channels (Channel status can be modeled with a Markov Chain [103]). Both nodes are equipped with multiple radios, which enable them to work on multiple channels simultaneously. However, due to channel fading, packet transmission between the pair may suffer from transmission failure. The successful packet transmission probability over ch_i is denoted as δ_i ($\delta_i = 1 - p_i$, p_i is the packet loss probability over ch_i). Furthermore we consider the system works in a slotted way, and the size of packets transmitted between node S and node D is deterministic. The duration of each slot is denoted as T. We assume that node S can transmit one packet to node D over each channel in each slot and an ACK message can be received, if necessary, in the same slot.

We want to determine the *batch delay*, which is defined as the time starting from the first attempt to look for available channels to transmit packets in this batch until all of the N packets in this batch (i.e. the entire batch) are successfully transmitted from node S to node D.

Batch transmission is a key component in network coding based opportunistic routing

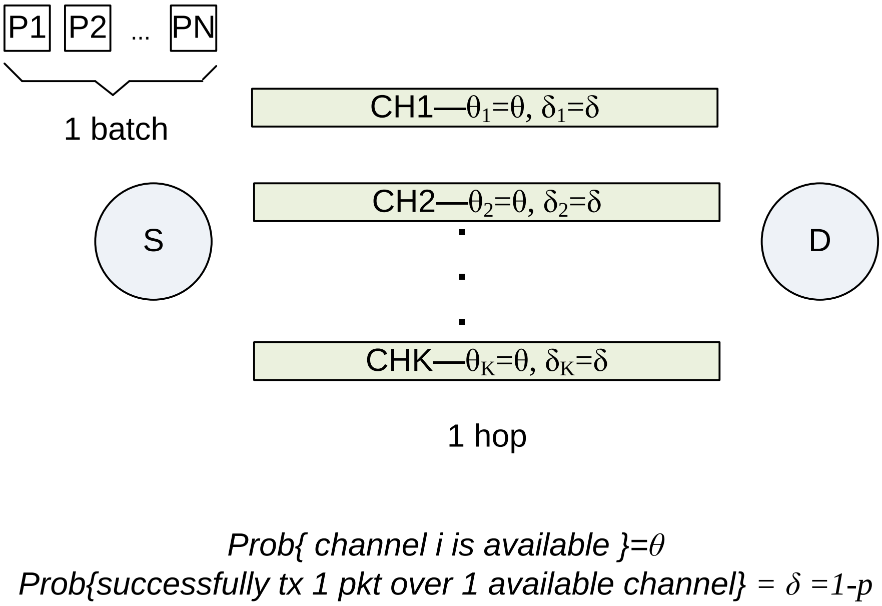


Figure 4.1: System model: N packets in a batch ready to be sent from S to D, with K potential channels in the channel pool

algorithms [39, 49]. The analysis of batch transmission over multiple channels between such a single-hop communication pair in a CRN, can provide an insight into how fast a batch of packets can be transmitted one hop in a multi-hop network. It also can provide a starting point for the performance analysis of network coding based opportunistic routing schemes. This single-hop analysis will be extended in future work to multi-hop networks to help design and get a better understanding of network coding based opportunistic routing in multi-channel CRNs.

Before analysing the batch delay, we need to describe how the N packets could be transmitted over a single-hop multi-channel link in a CRN. This is a new challenge emerging with multiple channel scenarios. For a single channel scenario, we transmit one packet in each slot and retransmit the packet in the next slot if it is lost, until the transmission of the entire batch completes. However, when there are multiple channels available, there are several transmission schemes that can be considered.

In the following, three transmission schemes are discussed that use different transmission and retransmission strategies.

4.2.2 Multi-Channel Transmission Schemes

1) Multi-channel ARQ based with Single-copy Retransmission of Lost Packets Scheme (SC-MARQ)

In the first scheme, multi-channel ARQ [91, 92] is employed for batch transmission. Different packets are transmitted over different channels, and only single-copies of the lost packets will be retransmitted. This scheme is referred to as the *multi-channel ARQ based with single-copy retransmission of lost packets* scheme (*SC-MARQ*).

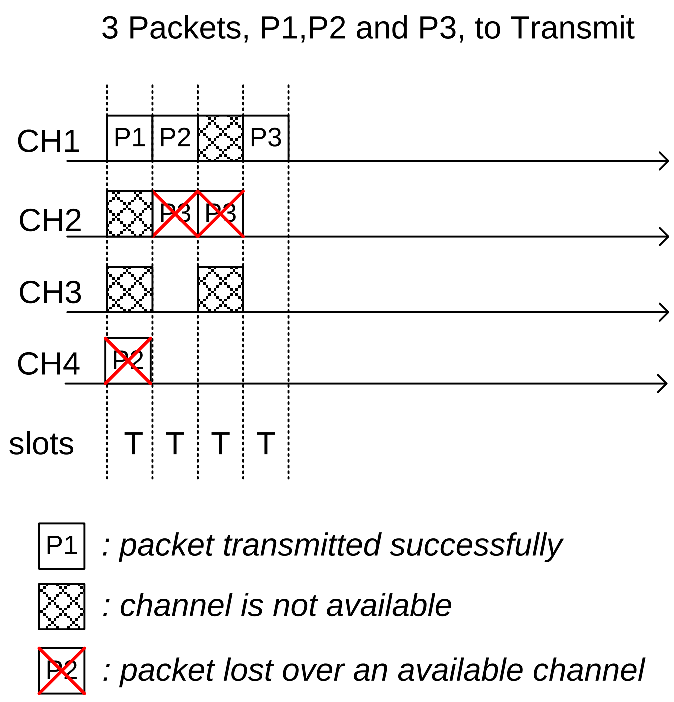


Figure 4.2: An illustration of SC-MARQ scheme

In each time slot, at source node S, k different packets are transmitted over k different available channels, and single-copies of the m lost packets will be retransmitted over m channels in the next time slot if m transmission failures happen. The scheme is illustrated as in Figure 4.2.

2) Multi-channel ARQ based with Multi-copy Retransmission of Lost Packets Scheme (MC-MARQ)

In the second scheme, different packets are transmitted over different channels; but lost packets together with packets left in the buffer will be transmitted over all the available channels, which mean some lost packets may be retransmitted with multiple duplicated copies. This scheme is referred to as the *multi-channel ARQ based with multi-copy retransmission of lost packets* scheme (*MC-MARQ*).

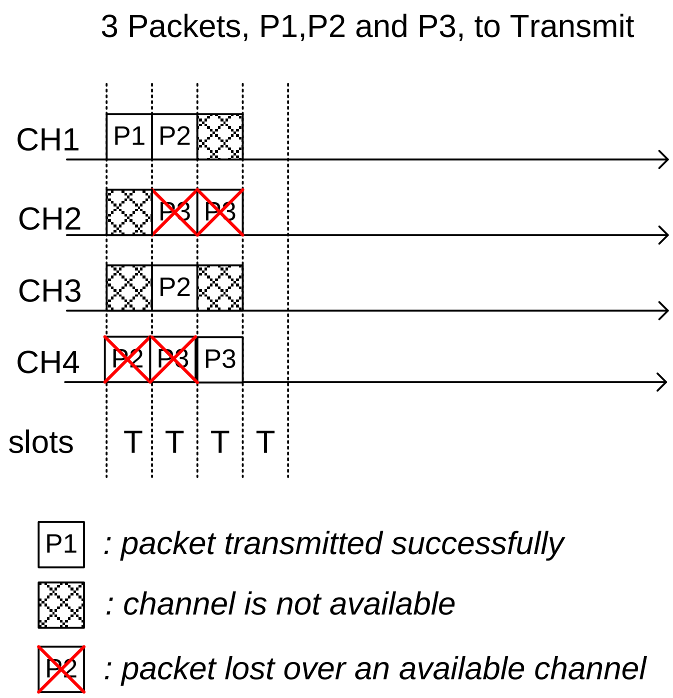


Figure 4.3: An illustration of MC-MARQ scheme

In each slot, at source node S, k different packets are transmitted over all the k different available channels, and all the k available channels will be used for retransmission of m lost packets (and packets are not transmitted yet). Some of the m lost packets could be retransmitted with multiple duplicated copies over multiple channels. Figure 4.3 illustrates this transmission scheme.

In this scheme, channel resources are fully exploited for retransmission of lost packets.

3) Random Linear Coding based Multi-channel Transmission Scheme (RLC-MCT)

In the third scheme, before transmission, the original packets in a batch are randomly coded as in [39]. Every coded packet is a random linear combination of the original packets in a batch. The coefficients of the combination are drawn uniformly from a finite field. The probability of selecting linearly dependent combinations is negligible over a sufficiently large field [104]. Then different coded packets are transmitted over different multiple available channels. The source node does not care about packet loss and it keeps coding new packets and transmitting them in each slot until it receives an ACK from the destination. The ACK will be sent from the destination when it gets enough (number of packets in a batch) independent packets to decode the whole batch. This scheme is referred to as the *random linear coding based multi-channel transmission* scheme (*RLC-MCT*).

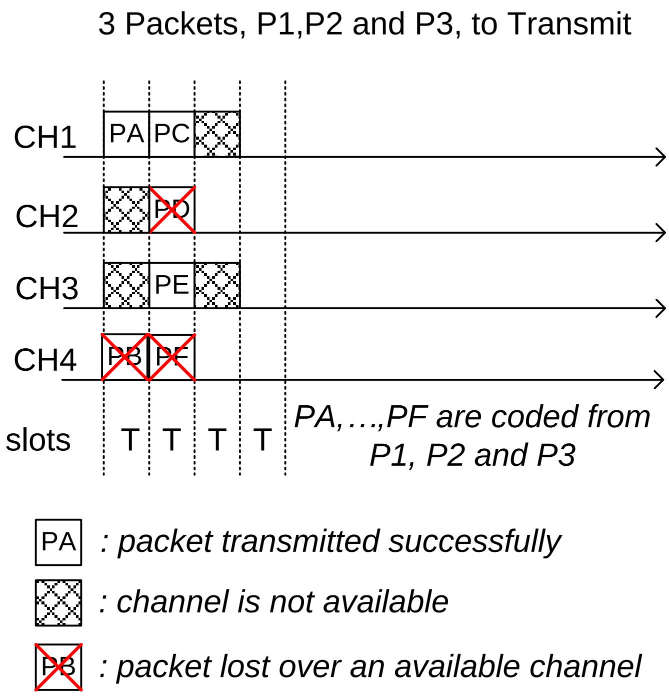


Figure 4.4: An illustration of RLC-MCT scheme

In each slot, at source node S, k different coded packets, coded from the N original packets, are transmitted over all the k different available channels. Node S will keep coding and transmitting k new packets in each time slot until it receives a batch ACK message from node D, which is sent when node D obtains N independent packets to decode the entire batch. Figure 4.4 shows an example of this scheme.

In essence, transmitting new coded packets for the same batch until a batch ACK is received by the source node is another way to achieve retransmission. This scheme utilizes channel resources more efficiently.

In SC-MARQ and MC-MARQ, destination node D sends an ACK message for every packet in the batch, but in RLC-MCT, destination node D only sends one ACK when it receives the entire batch.

4.3 Batch Delay Analysis for Different Schemes

Three different multi-channel schemes for batch transmission in CRNs have been introduced in the previous section. In this section, we will provide theoretical analysis of batch transmission delay associated with these three transmission schemes.

Working in CRNs, when source node S *attempts* to transmit packets to destination node D: S first checks how many channels are available and then transmits packets according to different schemes, given these available channels. We would like to first introduce four probabilities associated with the batch transmission process.

- $P_{(K)}^{(k)}$, the probability that k channels are available, given that there are K potential channels in the channel pool.
- $P_{i/n}^{(k)}$, the probability that destination node D receives *i effective packets* when source node S transmits *n effective packets* out to destination node D, given that k channels are available.

- $P_{i/k} = P_{i/k}^{(k)}$, when the number of *effective packets* is the same as the number of available channels, the superscript part for the number of available channels can be dropped.

- $P_{i,n}^{(K)}$, the probability that destination node D receives i *effective packets* when source node S attempts to transmits n *effective packets*, given that there are totally K potential channels in the channel pool.

"*Effective packets*" are judged by information carried by packets to the destination node. In SC-MARQ, all the k (or less) packets transmitted are different, they carry *k-packet information*, and thus they are counted as k *effective packets*; In MC-MARQ, if only m ($m \leq k$) different packets, but with (k-m) duplicated multiple copies of some of the m packets, are transmitted over k available channels, they carry *m-packet information*, and thus they are counted as m *effective packets*; In RLC-MCT, k coded packets are coded from N original packets. If the number of received packets at the destination is larger than (N-k), there must be some packets in the k coded packets linearly dependent with the received packets. Only the total number of packets independent to the received packets is counted, say J, the k packets carry *J-packet information* to the destination, and thus they are counted as *J effective packets*.

The probabilities $P_{(K)}^{(k)}$ and $P_{i/k}$ are given as:

$$P_{(K)}^{(k)} = C_K^k \theta^k (1-\theta)^{K-k} \tag{4.1}$$

$$P_{i/k} = C_k^i \delta^i (1-\delta)^{k-i} \tag{4.2}$$

The probabilities $P_{i/n}^{(k)}$ and $P_{i,n}^{(K)}$ depend on the different transmission schemes and are accordingly denoted as

$$P_{i/n}^{(k)} = \begin{cases} PA_{i/n}^{(k)}, & for\ SC-MARQ\ Scheme \\ PB_{i/n}^{(k)}, & for\ MC-MARQ\ Scheme \\ PC_{i/n}^{(k)}, & for\ RLC-MCT\ Scheme \end{cases} \tag{4.3}$$

$$P_{i,n}^{(K)} = \begin{cases} PA_{i,n}^{(K)}, & for\ SC-MARQ\ Scheme \\ PB_{i,n}^{(K)}, & for\ MC-MARQ\ Scheme \\ PC_{i,n}^{(K)}, & for\ RLC-MCT\ Scheme \end{cases} \tag{4.4}$$

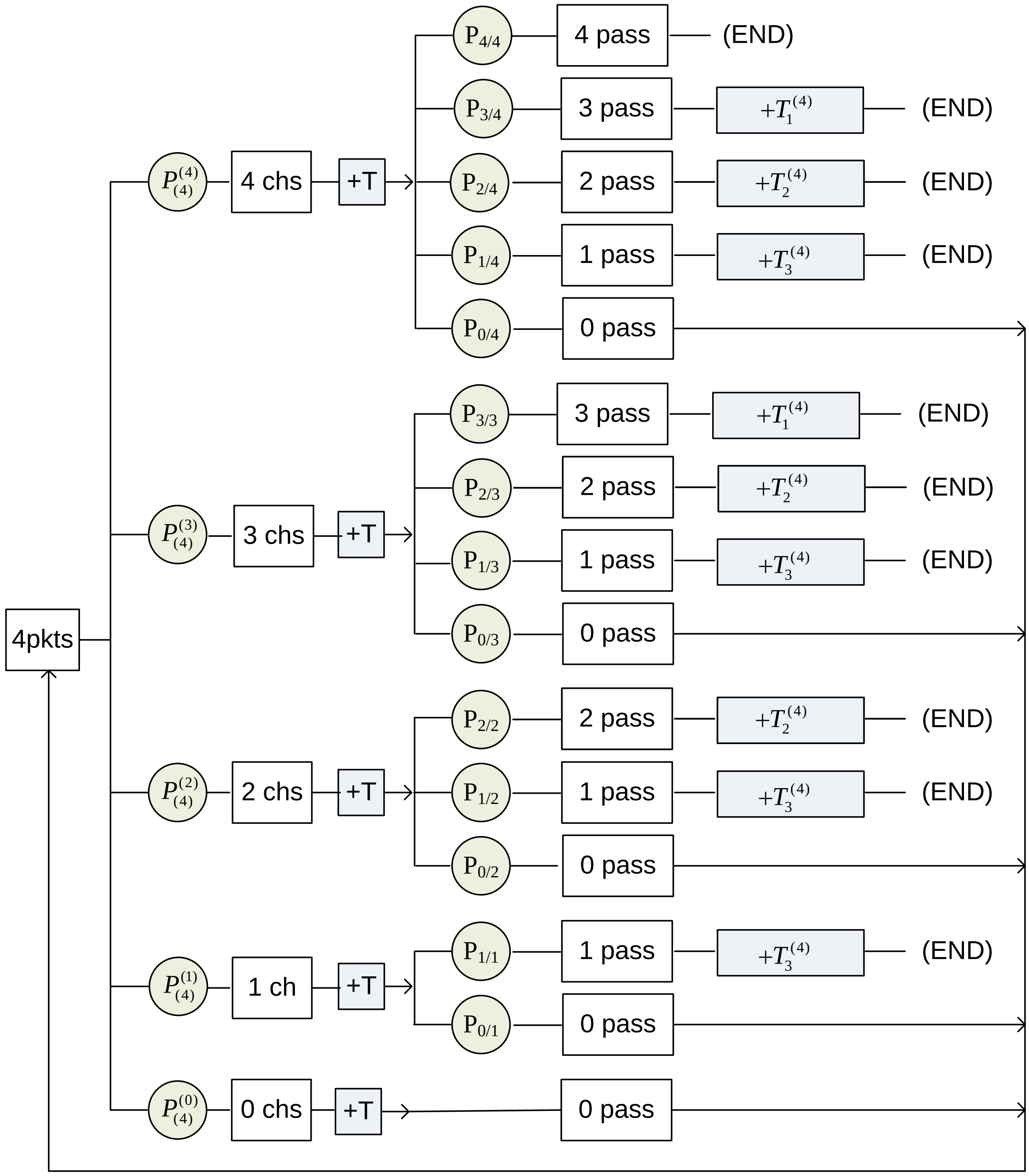
Moreover, $T_n^{(K)}$ denotes the delay that S transmits n packets to D, given that there are totally K potential channels in the channel pool. The transmission delays $T_n^{(K)}$ associated with different transmission scheme are denoted as

$$T_n^{(K)} = \begin{cases} TA_n^{(K)}, & for\ SC-MARQ\ Scheme \\ TB_n^{(K)}, & for\ MC-MARQ\ Scheme \\ TC_n^{(K)}, & for\ RLC-MCT\ Scheme \end{cases} \tag{4.5}$$

For a batch consisting of N packets, consequently, the batch delay is $T_N^{(K)}$.

4.3.1 Batch Delay in a General Form: n-packets-over-K-potential-channels

In this subsection, we provide a general form solution of the batch delay associated with the three transmission schemes. Then we derive related probabilities for the different schemes. The flow chart of the batch transmission process is described in Figure 4.5 with a case of 4-packet batch over 4 potential channels.



$P^{(k)}_{(K)}$: *Probability that k channels are available, given that there are totally K potential channels in the channel pool*

$P_{i/k} = P^{(k)}_{i/k}$: *Probability that destination receives i effective packets when source transmits k effective packets, given that k channels are available*

$T^{(K)}_{n}$: *Delay that source transmits n packets, given that there are totally K potential channels in the channel pool*

Figure 4.5: Transmission of 4-packet batch over 4 potential channels

When $n \leq K$, the transmission delay can be given as

$$\begin{aligned} T_n^{(K)} =& \frac{1}{1 - P_{0,n}^{(K)}} [P_{n,n}^{(K)} T \\ & + P_{n-1,n}^{(K)} (T + T_1^{(K)}) \\ & + P_{n-2,n}^{(K)} (T + T_2^{(K)}) \\ & ... \\ & + P_{2,n}^{(K)} (T + T_{n-2}^{(K)}) \\ & + P_{1,n}^{(K)} (T + T_{n-1}^{(K)})] + \frac{P_{0,n}^{(K)}}{1 - P_{0,n}^{(K)}} T \end{aligned} \tag{4.6}$$

where $T_1^{(K)} = \frac{1}{1-P_{0,1}^K} T$.

When $n > K$, the transmission delay can be given as

$$\begin{aligned} T_n^{(K)} =& \frac{1}{1 - P_{0,K}} [P_{K,K} (T + T_{n-K}^{(K)}) \\ & + P_{K-1,K} (T + T_{n-(K-1)}^{(K)}) \\ & + P_{K-2,K} (T + T_{n-(K-2)}^{(K)}) \\ & ... \\ & + P_{2,K} (T + T_{n-2}^{(K)}) \\ & + P_{1,K} (T + T_{n-1}^{(K)})] + \frac{P_{0,K}}{1 - P_{0,K}} T \end{aligned} \tag{4.7}$$

where $P_{i,K} = P_{i,K}^{(K)}$.

For $n \leq K$, we can derive a general form for $P_{i,n}^{(K)}$ as:

$$P_{i,n}^{(K)} = \sum_{k=n}^{K} P_{(K)}^{(k)} P_{i/n}^{(k)} + \sum_{k=i}^{n-1} P_{(K)}^{(k)} P_{i/k}^{(k)} \tag{4.8}$$

Due to $P_{i/k}^{(k)} = P_{i/k}$, it can be obtained via equation (4.2). We get $P_{i,n}^{(K)}$ via equation

(4.8) and further get $T_n^{(K)}$ via equations (4.6) and (4.7). Now we need to derive $P_{i/n}^{(k)}$ in the first item of the right part in equation (4.8) for $n \leq k \leq K$.

For SC-MARQ:

In this scheme, when there are n packets, they are transmitted on n available channels without any duplicated packets. Therefore, the probability that destination node D receives i effective packets when source node S transmits n packets can be calculated as below.

$$P_{i/n}^{(k)} = PA_{i/n}^{(k)} = C_n^i {P_{1/1}}^i {P_{0/1}}^{n-i} = C_n^i \delta^i (1-\delta)^{n-i} \tag{4.9}$$

For MC-MARQ:

When we have n packets and k available channels ($n \leq k$), we allocate channels for packets in a balanced way, the s^{th} packet will be transmitted over k_s channels.

Let $r = k \bmod n$. If $r = 0$, we have $k_s = k/n$. To receive i effective packets, the destination node has to receive i different packets. For each effective packet, it has k/n duplicated copies transmitted. The effective packet will be successfully received if any of the k/n copies is received. Thus we have

$$P_{i/n}^{(k)} = PB_{i/n}^{(k)} = C_n^i \left(PB_{1/1}^{(k/n)}\right)^i \left(PB_{0/1}^{(k/n)}\right)^{n-i} \tag{4.10}$$

where $PB_{0/1}^{(k/n)} = {P_{0/1}}^{k/n} = (1-\delta)^{k/n}$ and $PB_{1/1}^{(k/n)} = 1 - {P_{0/1}}^{k/n} = 1-(1-\delta)^{k/n}$.

If $r > 0$, we have the number of available channels that are allocated to the s^{th} packet,

$$k_s = \begin{cases} \left\lfloor \frac{k}{n} \right\rfloor + 1, & 1 \leq s \leq r \\ \left\lfloor \frac{k}{n} \right\rfloor, & s > r \end{cases} \tag{4.11}$$

For packets 1, 2, ..., r, they are transmitted in k_1 copies, while packets r+1, r+2, ..., n are transmitted in k_n copies. For the i effective packets, they can be a combination of j packets from the first r packets and i-j packets from the last n-r packets. The value range of j depends on the relationship among i, r, and n-r. And thus we have $P_{i/n}^{(k)}$ as follows.

(1) $i \leq \min(r, n-r)$. In this case, all the i packets can be chosen among packets 1,...,r, or packets r+1,..., n. Therefore we have the range of j, $0 \leq j \leq i$.

$$P_{i/n}^{(k)} = PB_{i/n}^{(k)} = \sum_{j=0}^{i} [C_r^j (PB_{1/1}^{(k_1)})^j (PB_{0/1}^{(k_1)})^{r-j}][C_{n-r}^{i-j} (PB_{1/1}^{(k_n)})^{i-j} (PB_{0/1}^{(k_n)})^{n-r-(i-j)}] \quad (4.12)$$

(2) $\min(r, n-r) < i \leq \max(r, n-r)$. In this case, the i packets cannot be chosen only from the group with the smaller number of packets. There are two subcases in this case.

a) $r < n-r$, then $r < i \leq n-r$. In this case, there are at most r of the i packets from the first r packets.

$$P_{i/n}^{(k)} = PB_{i/n}^{(k)} = \sum_{j=0}^{r} [C_r^j (PB_{1/1}^{(k_1)})^j (PB_{0/1}^{(k_1)})^{r-j}][C_{n-r}^{i-j} (PB_{1/1}^{(k_n)})^{i-j} (PB_{0/1}^{(k_n)})^{n-r-(i-j)}] \quad (4.13)$$

b) $r > n-r$, then $n-r < i \leq r$. In this case, there are at least i-(n-r) packets of the packets from the first r packet, considering that there are at most n-r packets from the last n-r packets.

$$P_{i/n}^{(k)} = PB_{i/n}^{(k)} = \sum_{i-(n-r)}^{i} [C_r^j (PB_{1/1}^{(k_1)})^j (PB_{0/1}^{(k_1)})^{r-j}][C_{n-r}^{i-j} (PB_{1/1}^{(k_n)})^{i-j} (PB_{0/1}^{(k_n)})^{n-r-(i-j)}] \quad (4.14)$$

(3) $\max(r, n-r) < i \leq n$. In this case, neither the first r packets nor the last n-r packets can provide all the i packets. So there are at least i-(n-r) and at most r packets from the first r packets.

$$P_{i/n}^{(k)} = PB_{i/n}^{(k)} = \sum_{i-(n-r)}^{r} [C_r^j (PB_{1/1}^{(k_1)})^j (PB_{0/1}^{(k_1)})^{r-j}][C_{n-r}^{i-j} (PB_{1/1}^{(k_n)})^{i-j} (PB_{0/1}^{(k_n)})^{n-r-(i-j)}] \quad (4.15)$$

For probabilities $PB_{0/1}^{(k_1)}$, $PB_{1/1}^{(k_1)}$, $PB_{0/1}^{(k_n)}$, and $PB_{1/1}^{(k_n)}$ in equations (4.12)-(4.15), they can be calculated as follows.

$$PB_{0/1}^{(k_1)} = {P_{0/1}}^{k_1} = (1-\delta)^{k_1} \tag{4.16}$$

$$PB_{1/1}^{(k_1)} = 1 - {P_{0/1}}^{k_1} = 1 - (1-\delta)^{k_1} \tag{4.17}$$

$$PB_{0/1}^{(k_n)} = {P_{0/1}}^{k_n} = (1-\delta)^{k_n} \tag{4.18}$$

$$PB_{1/1}^{(k_n)} = 1 - {P_{0/1}}^{k_n} = 1 - (1-\delta)^{k_n} \tag{4.19}$$

<u>For RLC-MCT</u>:

If there are only n effective packets information carried in k coded packets, any $j \geq n$ coded packets received are equivalent to n effective packets while any $j < n$ coded packets received are j effective packets. Thus we have

$$P_{i/n}^{(k)} = PC_{i/n}^{(k)} = \begin{cases} PC_{n/n}^{(k)} = \sum\limits_{j=n}^{k} PC_{j/k}^{(k)} = \sum\limits_{j=n}^{k} P_{j/k}, \quad i = n \\ PC_{i/n}^{(k)} = P_{i/k}, \quad i < n \end{cases} \tag{4.20}$$

Now we have obtained $P_{i/n}^{(k)}$ for SC-MARQ, MC-MARQ and RLC-MCT, further we can get $P_{i,n}^{(K)}$ via equation (4.8) and finally we can get all the batch delay via equations (4.6) and (4.7), letting $n = N$.

4.3.2 Illustration of Batch Delay Derivation

In this part, we use two examples to illustrate and verify the derivation of the batch delay in the previous subsection.

1) Case I: 2-packets-over-4-potential-channels ($n \leq K$)

The transmission process of 2 packets over 4 potential channels can be described via the flow chart in Figure 4.6. For the analysis to obtain the batch delay in a similar form to equation (4.6), the flow chart in Figure 4.6 can be redrawn as in Figure 4.7.

When there are 2 packets to be transmitted, no matter which transmission scheme is employed, there are three results after one time slot:

a) The destination node receives 2 effective packets and the transmission is completed. The transmission uses a time of T.

b) The destination node only receives 1 effective packet and the other packet needs to be transmitted, which will take $T_1^{(4)}$.

c) The destination does not receive any packet and the transmission goes to the starting point and needs to transmit the 2 packets.

According to the flow chart in Figure 4.7 and the definition of the probability$P_{i,n}^{(K)}$, we have the transmission delay for 2 packets as

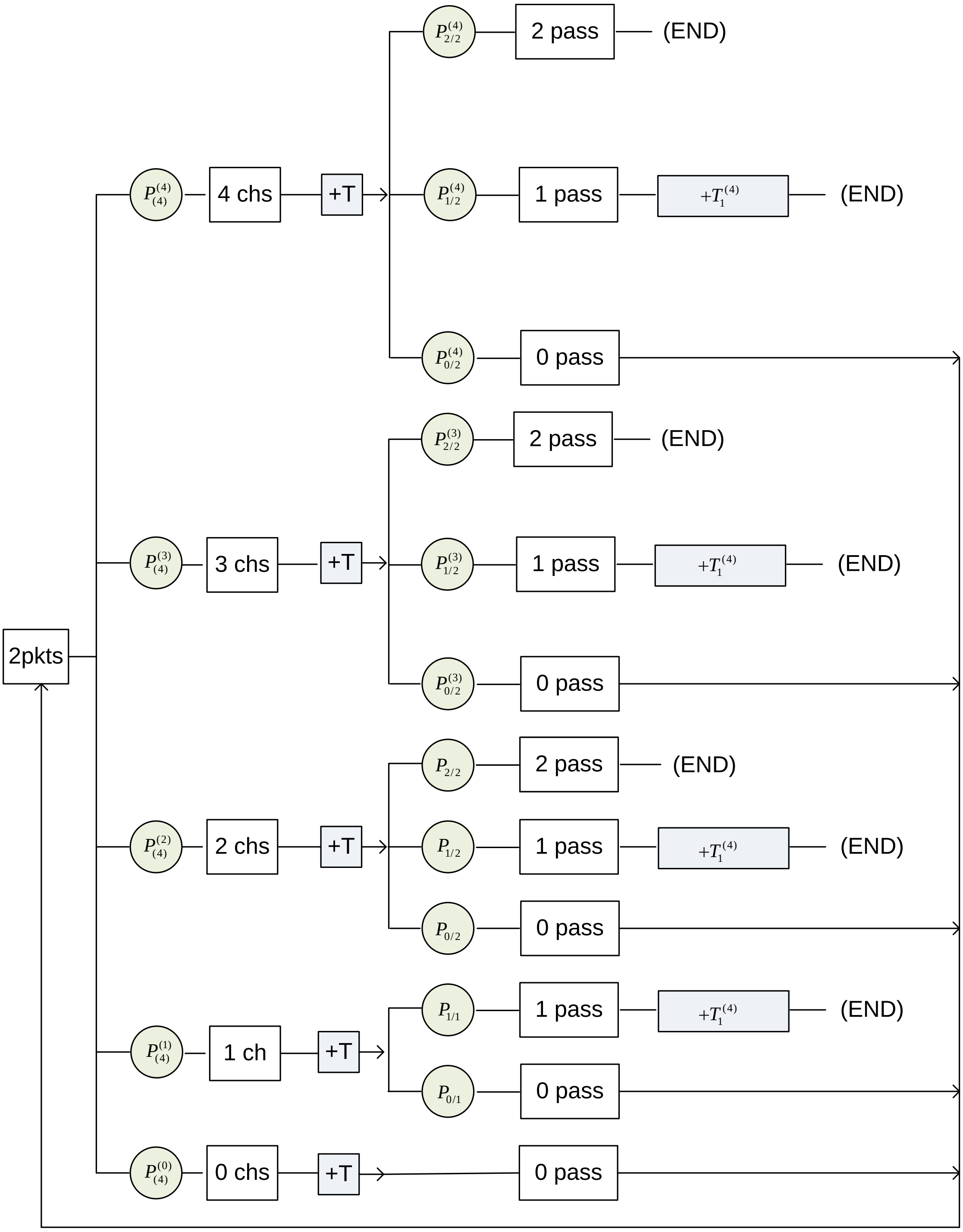
$$\begin{aligned} T_2^{(4)} &= [P_{2,2}^{(4)}T + P_{1,2}^{(4)}(T + T_1^{(4)})] \\ &\quad + P_{0,2}^{(4)}\{T + [P_{2,2}^{(4)}T + P_{1,2}^{(4)}(T + T_1^{(4)})] \\ &\quad\quad + P_{0,2}^{(4)}\{T + [P_{2,2}^{(4)}T + P_{1,2}^{(4)}(T + T_1^{(4)})] \\ &\quad\quad\quad + P_{0,2}^{(4)}\{T + [... \\ &= \frac{1}{1 - P_{0,2}^{(4)}}[P_{2,2}^{(4)}T \\ &\quad + P_{1,2}^{(4)}(T + T_1^{(4)})] + \frac{P_{0,2}^{(4)}}{1 - P_{0,2}^{(4)}}T \end{aligned} \tag{4.21}$$

The last equation in equation (4.21) is in a similar form to equation (4.6) for $n = 2$ and $K = 4$.

For $T_1^{(4)}$, we have

$$T_1^{(4)} = \frac{1}{1 - P_{0,1}^{(4)}}T = \frac{1}{P_{1,1}^{(4)}}T \tag{4.22}$$

where $P_{1,1}^{(4)} = P_{(4)}^{(4)}P_{1/1}^{(4)} + P_{(4)}^{(3)}P_{1/1}^{(3)} + P_{(4)}^{(2)}P_{1/1}^{(2)} + P_{(4)}^{(1)}P_{1/1}^{(1)}$. The probability $P_{(4)}^{(k)}$ $(k = 1, 2, 3, 4)$



$P_{(K)}^{(k)}$ *: Probability that k of the total K channels are available*

$P_{i/n}^{(k)}$ *: Probability that destination receives i effective packets when source transmits n effective packets, given that k channels are available*

$P_{i/k} = P_{i/k}^{(k)}$

$T_n^{(K)}$ *: Delay that source transmits n packets, given that there are K channels in channel pool*

Figure 4.6: Transmission of 2 packets over 4 potential channels

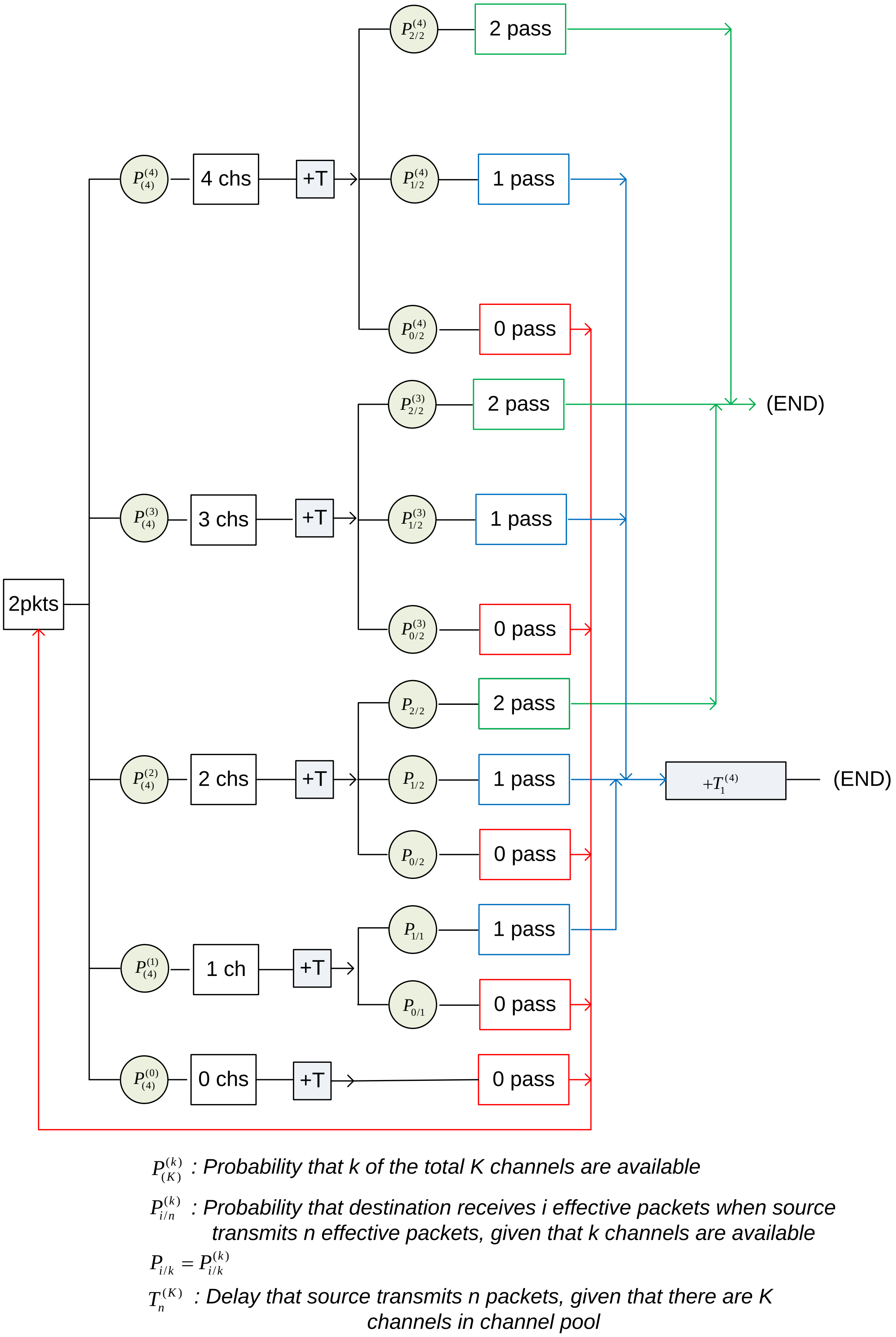


Figure 4.7: Analysis flow chart of 2 packets over 4 potential channels

can be calculated with equation (4.1) for $P^{(k)}_{(K)}$. The probability $P^{(k)}_{1/1}$ ($k = 1, 2, 3, 4$) for different transmission scheme can be calculated as follows.

For SC-MARQ,

$$P^{(4)}_{1/1} = P^{(3)}_{1/1} = P^{(2)}_{1/1} = P^{(1)}_{1/1} = P_{1/1} = \delta \tag{4.23}$$

For MC-MARQ,

$$P^{(k)}_{1/1} = PB^{(k)}_{1/1} = 1 - {P_{0/1}}^k = 1 - (1 - \delta)^k,\ k = 1, 2, 3, 4 \tag{4.24}$$

For RLC-MCT,

$$P^{(k)}_{1/1} = PC^{(k)}_{1/1} = 1 - {P_{0/1}}^k = 1 - (1 - \delta)^k,\ k = 1, 2, 3, 4 \tag{4.25}$$

By now, we can calculate the value of $T^{(4)}_1$ in equation (4.22) with the above equations. For the probability $P^{(4)}_{i,2}$ ($i = 0, 1, 2$) in equation (4.21), we can calculate it as follows.

$$\begin{aligned} P^{(4)}_{2,2} &= P^{(4)}_{(4)} P^{(4)}_{2/2} + P^{(3)}_{(4)} P^{(3)}_{2/2} + P^{(2)}_{(4)} P_{2/2} \\ &= P^{(4)}_{(4)} P^{(4)}_{2/2} + P^{(3)}_{(4)} P^{(3)}_{2/2} + P^{(2)}_{(4)} P^{(2)}_{2/2} \\ &= \sum_{k=2}^{4} P^{(k)}_{(4)} P^{(k)}_{2/2} \end{aligned} \tag{4.26}$$

In equation (4.26), the first line can be verified via the flow chart in Figure 4.7. The first line goes to the second line because $P_{i/k} = P^{(k)}_{i/k}$. The last line is in a form similar to equation (4.8) by subsisting $i = 2$, $n = 2$ and $K = 4$ (There is no second item because $i = n$.)

$$\begin{aligned} P^{(4)}_{1,2} &= P^{(4)}_{(4)} P^{(4)}_{1/2} + P^{(3)}_{(4)} P^{(3)}_{1/2} + P^{(2)}_{(4)} P_{1/2} + P^{(1)}_{(4)} P_{1/1} \\ &= P^{(4)}_{(4)} P^{(4)}_{1/2} + P^{(3)}_{(4)} P^{(3)}_{1/2} + P^{(2)}_{(4)} P^{(2)}_{1/2} + P^{(1)}_{(4)} P^{(1)}_{1/1} \\ &= \sum_{k=2}^{4} P^{(k)}_{(4)} P^{(k)}_{1/2} + \sum_{k=1}^{2-1} P^{(1)}_{(4)} P^{(k)}_{1/k} \end{aligned} \tag{4.27}$$

In equation (4.27), the first line can also be verified via the flow chart in Figure 4.7. The last line is in a form similar to equation (4.8) by subsisting $i = 1$, $n = 2$ and $K = 4$.

$$\begin{aligned} P_{0,2}^{(4)} &= P_{(4)}^{(4)} P_{0/2}^{(4)} + P_{(4)}^{(3)} P_{0/2}^{(3)} + P_{(4)}^{(2)} P_{0/2} + P_{(4)}^{(1)} P_{0/1} + P_{(4)}^{(0)} \\ &= P_{(4)}^{(4)} P_{0/2}^{(4)} + P_{(4)}^{(3)} P_{0/2}^{(3)} + P_{(4)}^{(2)} P_{0/2}^{(2)} + P_{(4)}^{(1)} P_{0/1}^{(1)} + P_{(4)}^{(0)} P_{0/0}^{(0)} \\ &= \sum_{k=2}^{4} P_{(4)}^{(k)} P_{0/2}^{(k)} + \sum_{k=0}^{2-1} P_{(4)}^{(k)} P_{0/k}^{(k)} \end{aligned} \tag{4.28}$$

In equation (4.28), the first line goes to the second line because $P_{0/0}^{(0)} = 1$. The last line is in a form similar to equation (4.8) by subsisting $i = 0$, $n = 2$ and $K = 4$.

In summary, with equations (4.26)-(4.28), we illustrated how equation (4.8) can be derived. Now we derive $P_{i/2}^{(k)}$ for $2 \leq k \leq 4$ associated with the different transmission schemes.

For SC-MARQ:

Since only single copies of different packets are transmitted, the 2 packets are transmitted over 2 channels as two effective packets. Thus we have

$$P_{2/2}^{(k)} = PA_{2/2}^{(k)} = PA_{2/2}^{(2)} = C_2^2 {P_{1/1}}^2 {P_{0/1}}^0 = C_2^2 \delta^2 (1-\delta)^0 \tag{4.29}$$

$$P_{1/2}^{(k)} = PA_{1/2}^{(k)} = PA_{1/2}^{(2)} = C_2^1 P_{1/1} P_{0/1} = C_2^1 \delta (1-\delta) \tag{4.30}$$

$$P_{0/2}^{(k)} = PA_{0/2}^{(k)} = PA_{0/2}^{(2)} = C_2^0 {P_{1/1}}^0 {P_{0/1}}^2 = C_2^0 \delta^0 (1-\delta)^2 \tag{4.31}$$

Equations (4.29)-(4.31) can be rewritten in a more general form as

$$P_{i/2}^{(k)} = PA_{i/2}^{(k)} = PA_{i/2}^{(2)} = C_2^i {P_{1/1}}^i {P_{0/1}}^{2-i} = C_2^i \delta^i (1-\delta)^{2-i} \tag{4.32}$$

Equation (4.32) is in a similar form to equation (4.9).

For MC-MARQ:

In this scheme, if there are more available channels than packets, some packets will be transmitted with duplicated copies.

For $k = 2$, the 2 packets are transmitted on the 2 channels, without any duplication. Hence we have,

$$P^{(2)}_{i/2} = PB^{(2)}_{i/2} = P_{i/2} = C^i_2 \delta^i (1-\delta)^{2-i} \tag{4.33}$$

For $k = 3$, the first packet is transmitted on 2 channels with duplication and the second packet is transmit on 1 channel.

$$\begin{aligned} P^{(3)}_{0/2} = PB^{(3)}_{0/2} &= PB^{(2)}_{0/1} PB^{(1)}_{0/1} \\ &= {P_{0/1}}^2 P_{0/1} \\ &= (1-\delta)^2 (1-\delta) \\ &= (1-\delta)^3 \end{aligned} \tag{4.34}$$

$$\begin{aligned} P^{(3)}_{1/2} = PB^{(3)}_{1/2} &= PB^{(2)}_{1/1} PB^{(1)}_{0/1} + PB^{(2)}_{0/1} PB^{(1)}_{1/1} \\ &= (1 - {P_{0/1}}^2) P_{0/1} + {P_{0/1}}^2 (1 - P_{0/1}) \\ &= (1 - (1-\delta)^2)(1-\delta) + (1-\delta)^2 \delta \end{aligned} \tag{4.35}$$

$$\begin{aligned} P^{(3)}_{2/2} = PB^{(3)}_{2/2} &= PB^{(2)}_{1/1} PB^{(1)}_{1/1} \\ &= (1 - {P_{0/1}}^2)(1 - P_{0/1}) \\ &= (1 - (1-\delta)^2)\delta \end{aligned} \tag{4.36}$$

For $k = 4$, each of the 2 packets are transmitted on 2 channels with duplication.

$$\begin{aligned} P^{(4)}_{i/2} = PB^{(4)}_{i/2} &= C^i_2 \left(PB^{(2)}_{1/1}\right)^i \left(PB^{(2)}_{0/1}\right)^{2-i} \\ &= C^i_2 \left(1 - {P_{0/1}}^2\right)^i \left({P_{0/1}}^2\right)^{2-i} \\ &= C^i_2 \left(1 - (1-\delta)^2\right)^i (1-\delta)^{2(2-i)} \end{aligned} \tag{4.37}$$

For RLC-MCT:

In this scheme, the 2 packets will be coded into k different packets and will be transmitted on k channels. With coding, any $j \leq 2$ coded packets are j effective packets. If there are more than 2 packets received, due to linear dependency, there are still only 2 effective packets.

$$P_{0/2}^{(k)} = PC_{0/2}^{(k)} = P_{0/k} \tag{4.38}$$

$$P_{1/2}^{(k)} = PC_{1/2}^{(k)} = P_{1/k} \tag{4.39}$$

$$P_{2/2}^{(k)} = PC_{2/2}^{(k)} = P_{2/k} + ... + P_{k/k} = \sum_{j=2}^{k} P_{j/k} \tag{4.40}$$

With equations (4.38)-(4.40), we have

$$P_{i/n}^{(k)} = PC_{i/n}^{(k)} = \begin{cases} PC_{2/2}^{(k)} = \sum_{j=2}^{k} P_{j/k}, & i = 2 \\ PC_{i/2}^{(k)} = P_{i/k}, & i < 2 \end{cases} \tag{4.41}$$

Equation (4.41) is in a similar form to equation (4.20).

2) Case II: 8-packets-over-4-potential-channels ($n > K$)

The transmission process of a 8-packet batch over 4 potential channels can be described via the flow chart in Figure 4.8. For the analysis to obtain the batch delay in a similar form to equation (4.7), the flow chart in Figure 4.8 can be redrawn as in Figure 4.9.

When there are 8 packets to be transmitted, no matter which transmission scheme is employed, at most 4 packets can transmitted over 4 channels and there are five results after one time slot:

a) The destination node receives 4 effective packets and the other 4 packet needs to be transmitted, which will take $T_4^{(4)}$.

b) The destination node only receives 3 effective packets and the other 5 packet needs to be transmitted, which will take $T_5^{(4)}$.

c) The destination node only receives 2 effective packets and the other 6 packet needs to be transmitted, which will take $T_6^{(4)}$.

d) The destination node only receives 1 effective packet and the other 8 packet needs to be transmitted, which will take $T_7^{(4)}$.

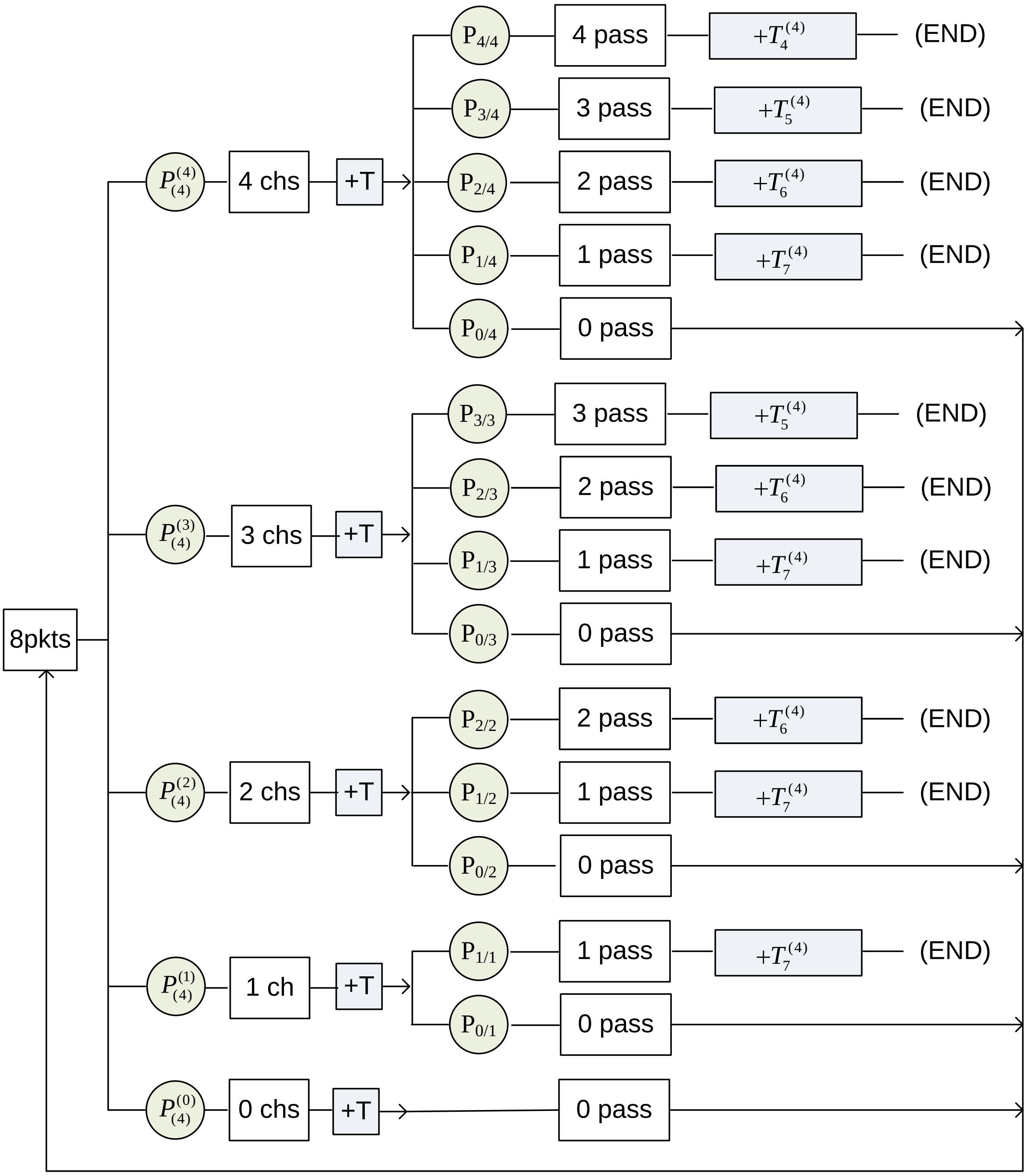
e) The destination does not receive any packet and the transmission goes to the starting point and needs to transmit the 8 packets.

According to the flow chart in Figure 4.9 and the definition of the probability $P_{i,n}^{(K)}$, similarly to case I, we have the transmission delay for 8 packets as

$$\begin{aligned} T_8^{(4)} = & [P_{4,4}^{(4)}(T+T_4^{(4)}) + P_{3,4}^{(4)}(T+T_5^{(4)}) + P_{2,4}^{(4)}(T+T_6^{(4)}) + P_{1,4}^{(4)}(T+T_7^{(7)})] \\ & + P_{0,4}^{(4)}\{T + [P_{4,4}^{(4)}(T+T_4^{(4)}) + P_{3,4}^{(4)}(T+T_5^{(4)}) + P_{2,4}^{(4)}(T+T_6^{(4)}) \\ & + P_{1,4}^{(4)}(T+T_7^{(7)})] \\ & + P_{0,4}^{(4)}\{T+[\ldots \\ = & \frac{1}{1-P_{0,4}^{(4)}}[P_{4,4}^{(4)}(T+T_4^{(4)}) \\ & + P_{3,4}^{(4)}(T+T_5^{(4)}) \\ & + P_{2,4}^{(4)}(T+T_6^{(4)}) \\ & + P_{1,4}^{(4)}(T+T_7^{(7)})] + \frac{P_{0,4}^{(4)}}{1-P_{0,4}^{(4)}}T \end{aligned} \tag{4.42}$$

Equation (4.42) is in a similar form to equation (4.7). Probability $P_{i,4}^{(4)}$ can be calculated with equation (4.8). Transmission delay $T_4^{(4)}$ can be calculated with equation (4.6), and then $T_5^{(4)}$, $T_6^{(4)}$, $T_7^{(4)}$ can be calculated in a similar way as equation (4.42).

In summary, through case I and case II, we have illustrated the derivation of the batch transmission delay for the different schemes.

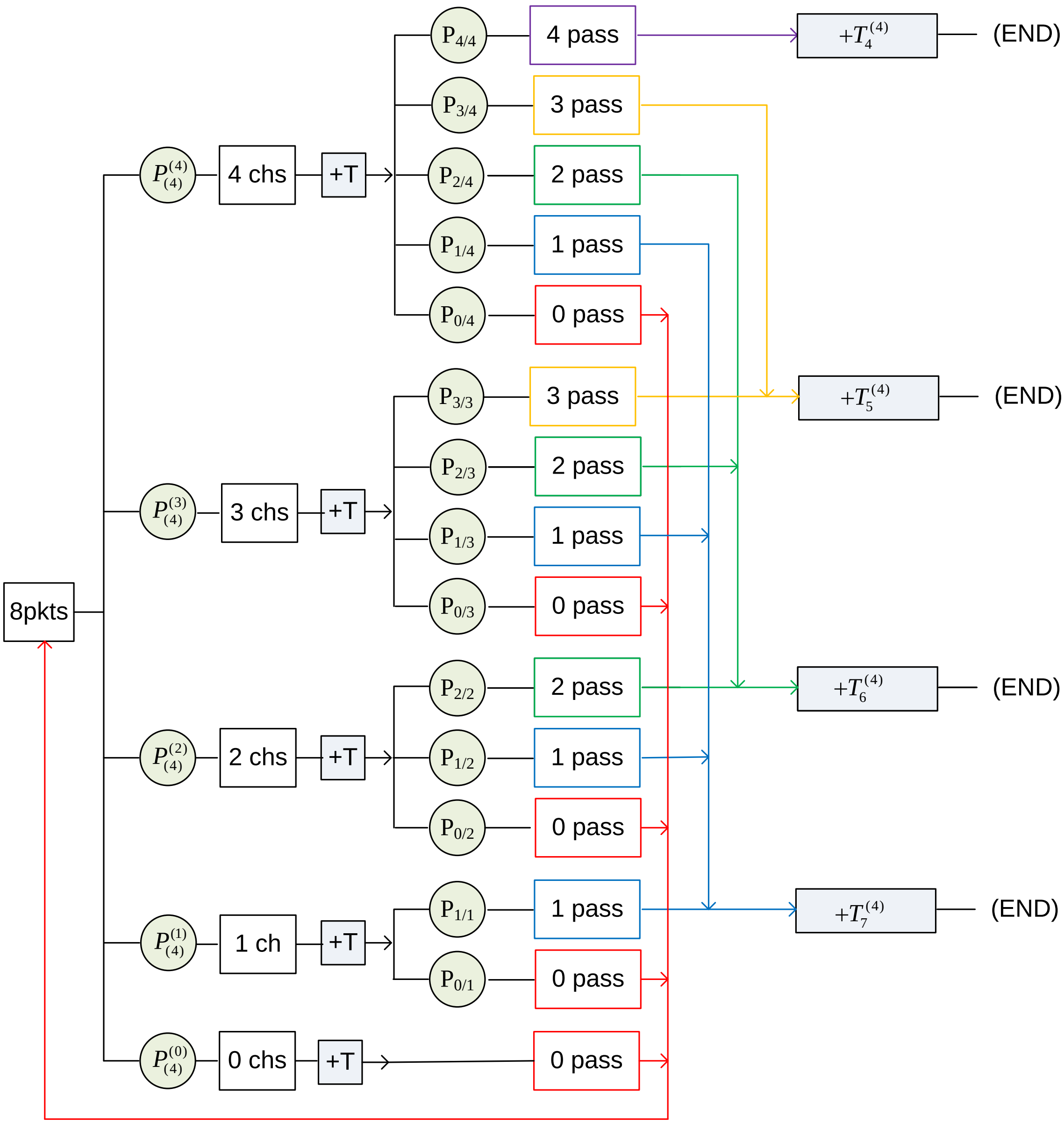


$P^{(k)}_{(K)}$: *Probability that k channels are available, given that there are totally K potential channels in the channel pool*

$P_{i/k} = P^{(k)}_{i/k}$: *Probability that destination receives i effective packets when source transmits k effective packets, given that k channels are available*

$T^{(K)}_n$: *Delay that source transmits n packets, given that there are totally K potential channels in the channel pool*

Figure 4.8: Transmission of 8 packets over 4 potential channels



$P_{(K)}^{(k)}$: *Probability that k channels are available, given that there are totally K potential channels in the channel pool*

$P_{i/k} = P_{i/k}^{(k)}$: *Probability that destination receives i effective packets when source transmits k effective packets, given that k channels are available*

$T_n^{(K)}$: *Delay that source transmits n packets, given that there are totally K potential channels in the channel pool*

Figure 4.9: Analysis flow chart of 8 packets over 4 potential channels

4.4 Performance Evaluation

In this section, we first verify the theoretical analysis presented in the last section using simulation. Then we compare performance of the three transmission schemes, SC-MARQ, MC-MARQ and RLC-MCT.

In the simulation, whether a channel is available in each slot is controlled via a random variable following the Bernoulli distribution with the probability of channel availability, θ; whether a packet, transmitted over an available channel, can be received successfully in each slot is controlled via another random variable following the Bernoulli distribution with the packet successful transmission probability, δ. In each slot, the source node first checks which channels are available; then it will transmit packets over the available channels under a chosen transmission scheme; at the end of each slot, it checks whether the packets are successfully received. In SC-MARQ and MC-MARQ schemes, packets are ACKed for every packet in a batch, while in RLC-MCT, packets are ACKed only when the entire batch of packets is received. The simulation is conducted with Matlab programming.

4.4.1 Theoretical Analysis Verification via Simulation

To verify the theoretical analysis, we simulated batch transmissions with the three schemes in two scenarios (different batch sizes) with different probabilities of channel availability: 1) 2-packet-batch-over-4-potential-channels (i.e. $N \leq K$), with probability of channel availability set to be 1, 0.5, and 0.2; 2) 8-packet-batch-over-4-potential-channels (i.e. $N > K$) and with probability of channel availability also set to be 1, 0.5, and 0.2.

The results for both scenarios are shown in Figures 4.10-4.12 and Figures 4.13-4.15 respectively. Also, we simulated these schemes in other batch sizes and other numbers of potential channels but the results are not provided here due to limited space.

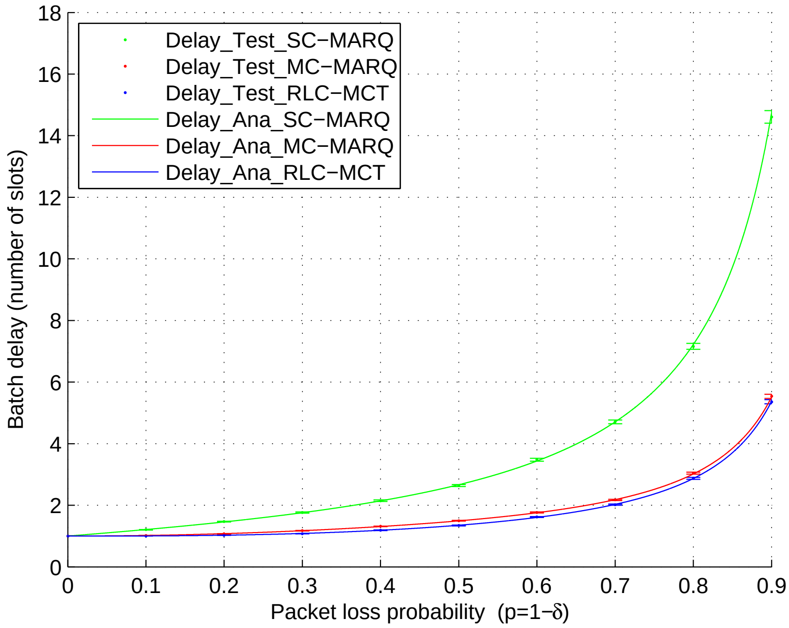


Figure 4.10: Batch delay as a function of packet loss probability with different probabilities of channel availability: 2-packet-batch-over-4-potential-channels, $\theta = 1$

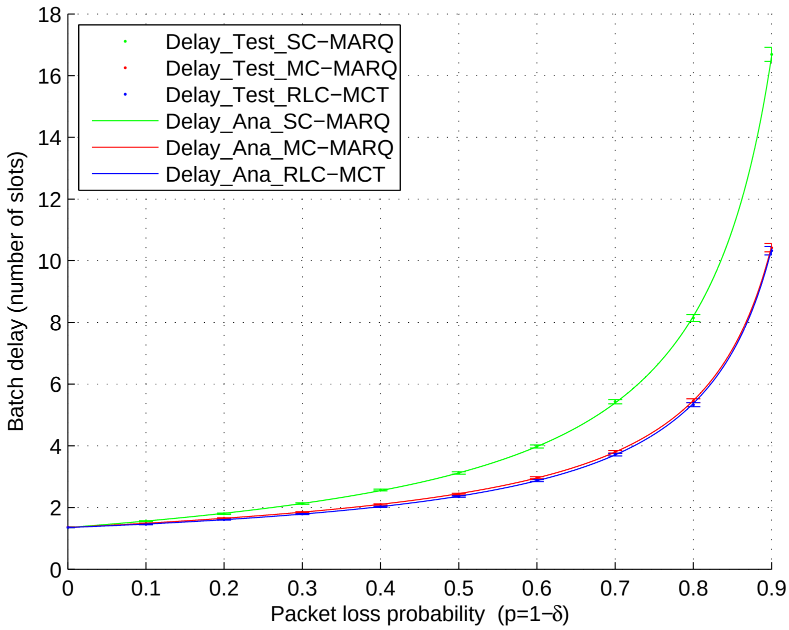


Figure 4.11: Batch delay as a function of packet loss probability with different probabilities of channel availability: 2-packet-batch-over-4-potential-channels, $\theta = 0.5$

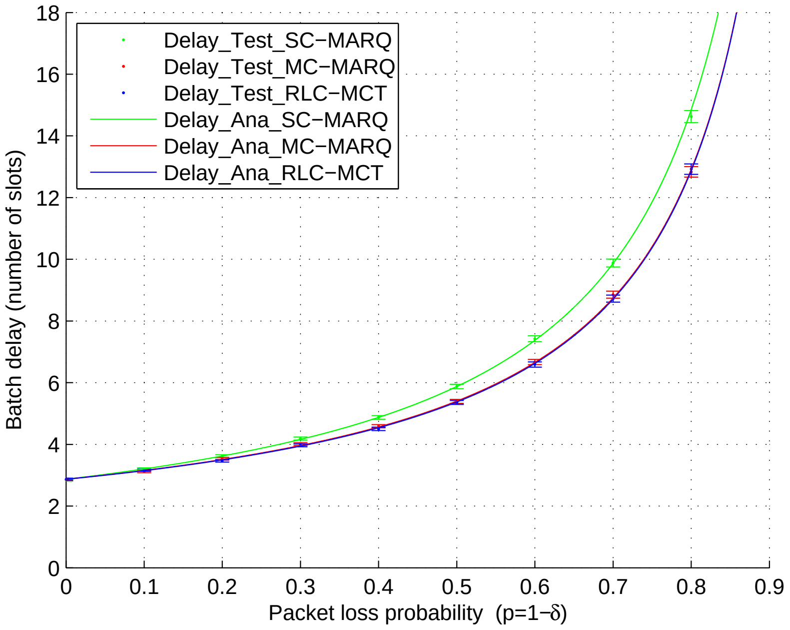


Figure 4.12: Batch delay as a function of packet loss probability with different probabilities of channel availability: 2-packet-batch-over-4-potential-channels, $\theta = 0.2$

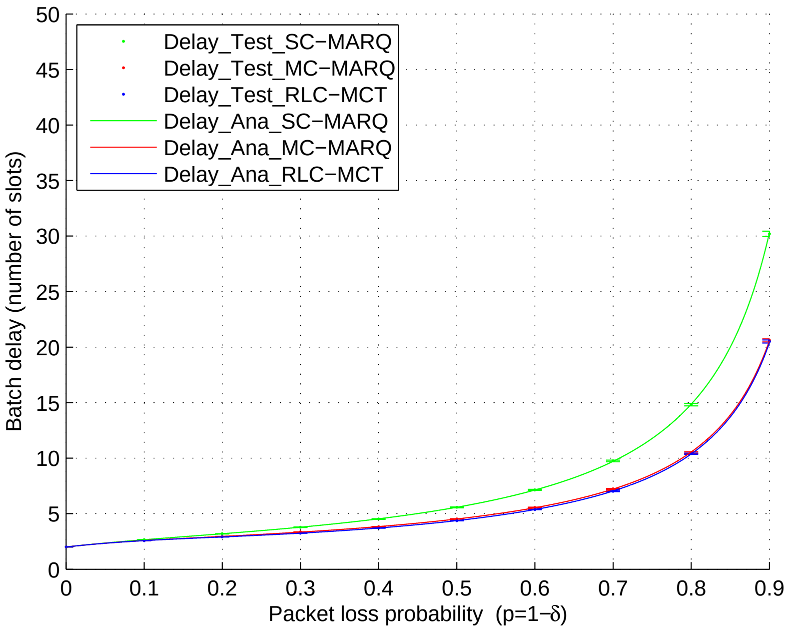


Figure 4.13: Batch delay as a function of packet loss probability with different probabilities of channel availability: 8-packet-batch-over-4-potential-channels, $\theta = 1$

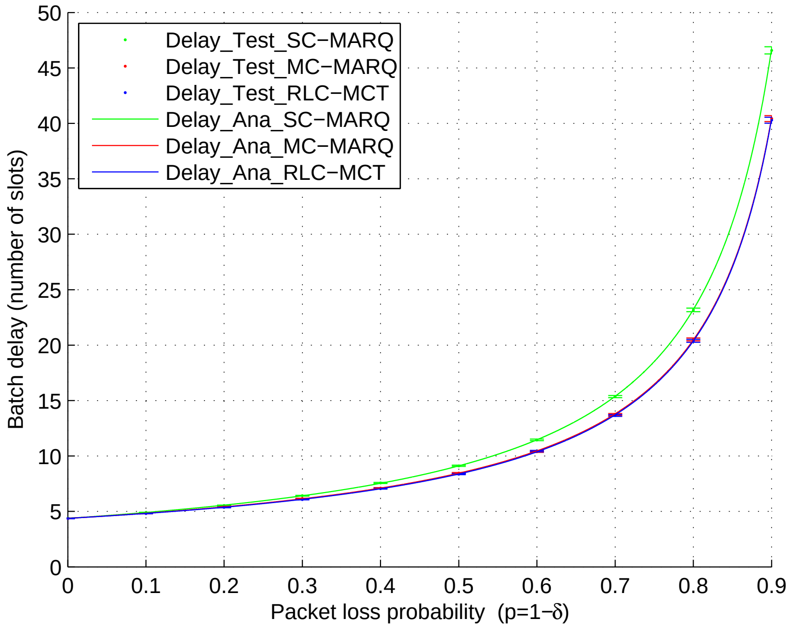


Figure 4.14: Batch delay as a function of packet loss probability with different probabilities of channel availability: 8-packet-batch-over-4-potential-channels, $\theta = 0.5$

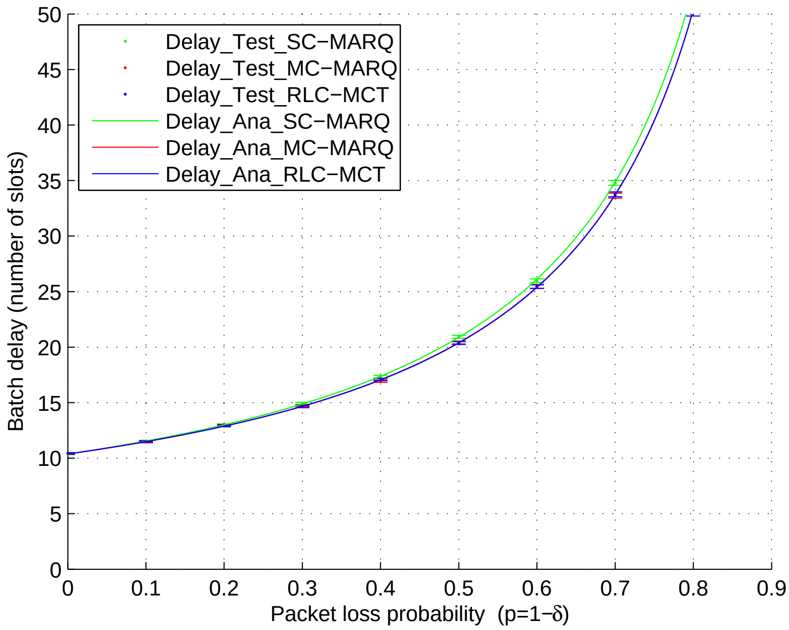


Figure 4.15: Batch delay as a function of packet loss probability with different probabilities of channel availability: 8-packet-batch-over-4-potential-channels, $\theta = 0.2$

The simulation results from all the scenarios with different settings match well with the analysis. In Figures 4.10-4.12 and Figures 4.13-4.15, it is shown that the 95% confidence intervals are very small and all the analyzed values are covered by the corresponding confidence intervals of the simulated values.

4.4.2 Performance Comparison among SC-MARQ, MC-MARQ and RLC-MCT

In general, it is shown that, the batch delay performance of RLC-MCT outperforms that of the other two schemes, SC-MARQ and MC-MARQ, based on all the results from both batch size scenarios with different probabilities of channel availability and different probabilities of packet successful transmission, in Figures 4.10-4.12 and Figures 4.13-4.15, except when the channel is perfect (Packet loss probability is 0).

In Figures 4.10-4.12, to make the comparison clearer, all the curves are drawn with the same ranges for the X axis and the Y axis.

In Figure 4.10, it can be seen that, the batch delay performance gap between RLC-MCT and SC-MARQ is obvious and it gets larger as packet loss probability increases from 0 to 0.9. The reason behind this is that, RLC-MCT retransmits lost packets (via transmitting new coded packets) fully exploiting multiple channels, while SC-MARQ only retransmits single copies of m lost packet. There are very few multi-channel benefits for retransmission under SC-MARQ, which becomes more obvious with worse channel conditions.

The gap between RLC-MCT and MC-MARQ is small, but RLC-MCT is always better than MC-MARQ. When m lost packets are retransmitted, some of the lost packets are retransmitted with multiple duplicated copies, which makes the retransmission more robust. This is why MC-MARQ outperforms SC-MARQ.

However, any m of the k packets are different under RLC-MCT, but there may be duplicated copies in any m of the k packets under MC-MARQ in each slot during retransmission. Thus the number of effective packets received in MC-MARQ could be less than m, even when the number of packets received is larger than m; but the number of effective packets received in RLC-MCT is m if the number of packets received is not less than m. In spite of the small delay gap between the two schemes, RLC-MCT requires only one ACK for a batch while MC-MARQ requires an ACK for every packet in the batch.

Similar results can be observed in Figure 4.11 and Figure 4.12. Moreover, when there are fewer available channels, the difference among all the packets transmitted becomes smaller and thus the gaps between the different schemes also become smaller.

Similar results can be found in Figures 4.13-4.15 as in Figures 4.10-4.12, but with a larger batch delay due to a bigger batch size.

In a summary, the analytical results are validated by the simulation. The batch delay performance of RLC-MCT outperforms that of SC-MARQ and MC-MARQ. Moreover, RLC-MCT requires only one ACK for a batch while SC-MARQ and MC-MARQ require ACKs for every packet in the batch.

4.5 Summary

In this chapter, we proposed a random linear coded scheme for efficient data transmission in multi-channel CRNs. We provided theoretical analysis and derived general form solutions of the batch delay associated with the proposed scheme and two multi-channel ARQ based schemes in a single-hop CRN. Analytical results, validated by simulations, show that batch delay of the coded scheme (RLC-MCT) is lower than that of the ARQ based schemes, under different probabilities of channel availability and packet loss probabilities. Moreover RLC-MCT only requires one ACK for the entire batch of packets, while the ARQ based schemes need an ACK for every packet. Such low dependence on feedback channels makes RLC-MCT more suitable to work in the dynamic spectrum environment in CRNs.

In the next chapter, we are going to extend such multi-channel batch transmission schemes to two-hop CRNs and make further analysis for the two-hop transmission in CRNs working in highly dynamic spectrum environments.

Chapter 5

Network Coding Based Transmission in Two-Hop Multi-Channel Cognitive Radio Networks

5.1 Overview

In cognitive radio networks, there is no guarantee on the availability of spectrum for the opportunistic communications between cognitive radio users. Transmission design plays an important role in leveraging the benefits of CRNs. In previous chapters, an opportunistic routing algorithm was designed for multi-hop data transmission in multi-channel CRNs with fast varying spectrum availability. A key component in the proposed opportunistic routing algorithm is batch transmission, where packets are transmitted in group (batch) from the source node. In a transmitted batch of packets, no packets can be recovered until the entire batch of packets is received by the destination node. The performance of batch transmission is crucial to the performance of the opportunistic routing algorithm for the multi-hop data transmission. Therefore, batch transmission should be well designed and the performance of batch transmission should be carefully examined in a dynamic cognitive radio environment.

In this chapter, we focus on data transmission in 2-hop multi-channel CRNs with lossy channels. We propose an efficient batch transmission scheme, the Opportunistic Routing combining Different packets over Multi-channel with network Coding (OR_DM-C) scheme. The OR_DM-C scheme exploits both packet forwarding and overhearing for data transmission by broadcasting different coded packets over multiple dynamically available channels. We derive network performance measures in terms of batch transmission delay for the proposed scheme. Batch transmission delay provides a better insight into the data transmission capability for network coding based opportunistic routing algorithm than simple packet link delay, because no packets can be processed before the entire batch is received by the destination node. The batch delay analysis in a 2-hop CRN provides initial insights into the more complicated CRNs. Two other schemes are also analyzed for comparison, namely, Opportunistic Routing combining Same packet over Multi-channel (OR_SM) scheme, which is employed in MSAOR [85], and a Traditional Routing strategy combining Different packets over Multi-channel with Multi-copy retransmission (TR_DM-M) scheme. Our analyses, validated by simulations, show that OR_DM-C outperforms the other two schemes in various network settings, and it reduces batch delay by up to 60%.

The rest of this chapter is organized as follows. In Section 5.2, we introduce the system model and the three 2-hop transmission schemes. The batch transmission delay associated with these three transmission schemes is derived in Section 5.3. In Section 5.4 the performance of these schemes is compared using validated analytical results. The work is summarized in the last section.

5.2 System Model and Transmission Schemes

5.2.1 System Model

We consider a 2-hop multi-channel CRN in a lossy wireless environment such as in Figure 5.1. There are three CR nodes in the network: S, R and D. There is a batch of N packets in the buffer of the source node S ready to be sent to the destination node D. Node R can serve as a forwarder for the packets transmitted from node S to node D.

We assume that there are totally K potential channels $\{ch_1, ch_2, ..., ch_K\}$ in the system. However, due to primary users' activities, these potential channels may be not available and the probability that a channel ch_i is available is denoted as θ_i . Thus in each time slot the number of available channels can be denoted as k ($k \in \{0, 1, ..., K\}$), which could be different in different time slots. In addition, due to channel fading, transmission between nodes can experience failure. The transmission success probabilities between nodes S and R, R and D over ch_i are denoted as $\delta_{SR}^{(i)}$ and $\delta_{RD}^{(i)}$, respectively. Moreover, packets transmitted from node S to node R on ch_i may be overheard by node D with a probability of $\delta_{SD}^{(i)}$.

Each CR node is equipped with multiple radios which enable CR nodes to work on multiple channels at the same time. Each radio works in a half-duplex way over a channel and thus nodes can only either transmit or receive packets on a given available channel. A packet received in a slot cannot be transmitted in the same slot.

We assume that the CRN works in a slotted way. The duration of each time slot is T. The length of the packets transmitted in the CRN is deterministic and a CR node can transmit only one packet in a slot if it successfully accesses an available channel.

Our goal is to determine how long the batch (transmission) delay is. The *batch (transmission) delay* is defined as the time starting from the first attempt of the source node to look for an available channel to transmit packets in this batch until the entire batch

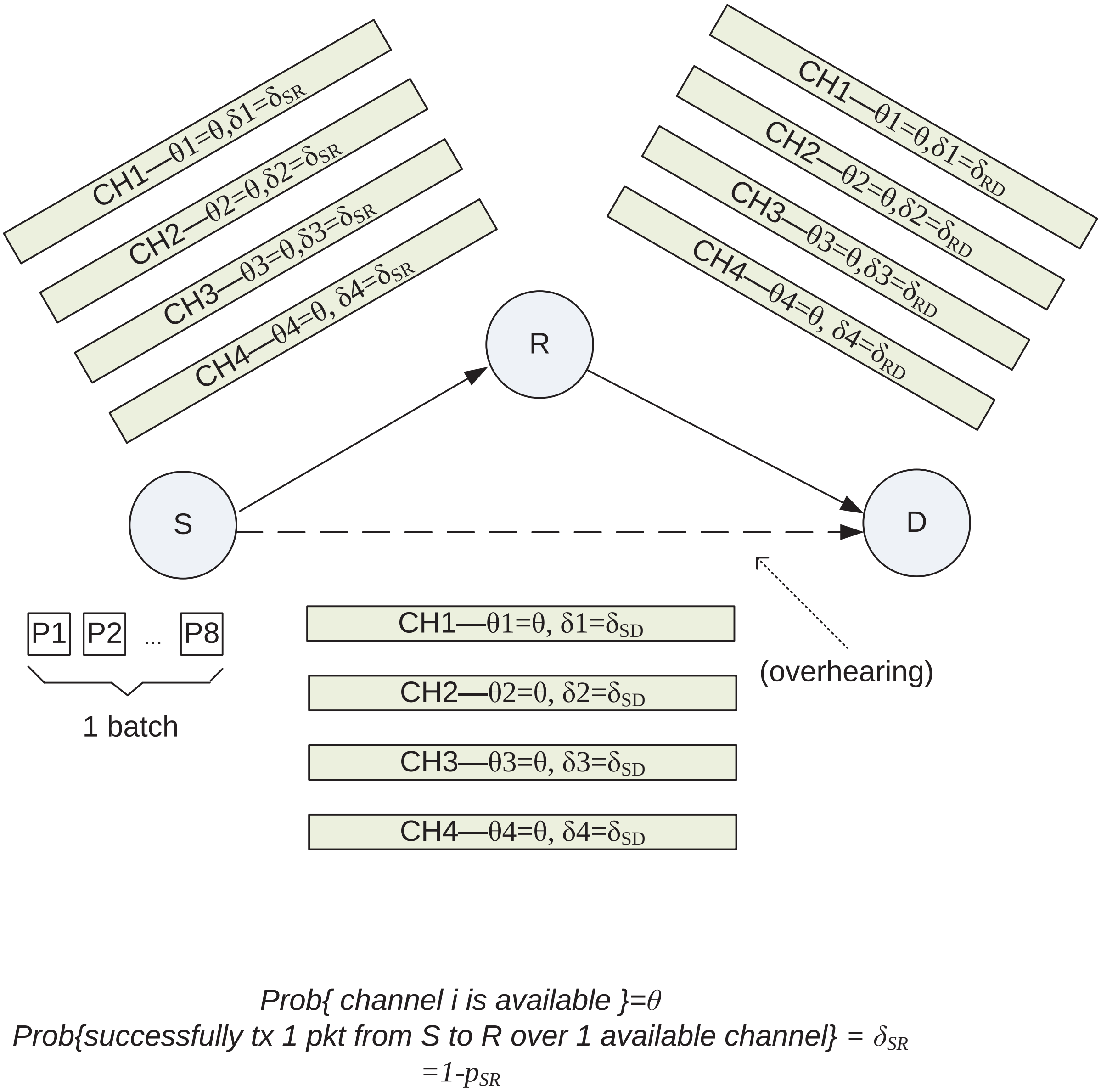


Figure 5.1: System model: 8-packet batch ready to be sent from S to D via R. Node D may overhear the packets sent from S to R. There are 4 potential channels, and their availabilities depend on primary users' activities. Packet transmission between two nodes over one channel is with a given transmission success probability.

(i.e. all N packets in the batch) is received by the destination node.

Before investigating batch delay, we would like to introduce three different data transmission schemes in single-hop multi-channel CRNs. Then these single-hop transmission schemes will be extended to 2-hop multi-channel CRNs, combined with different routing strategies.

5.2.2 Single-Hop Multi-Channel Data Transmission Schemes

1) Same Packet over Multi-Channel (SM)

In this scheme, multiple copies of the same packet are attempted to transmit over all available channels between the source node and the destination node within one hop. This scheme is referred to as same packet over multi-channel scheme (SM). In this scheme, a packet can be transmitted in minimum time compared to the single channel scheme.

2) Different Packets over Multi-Channel with Multi-Copy Retransmission of Lost Packets (DM-M)

In this scheme, different packets are transmitted over different available channels. If some packets are lost in one slot, they will be retransmitted in the next slot over all the available channels. This means that some of these lost packets may be retransmitted with multiple copies. This scheme is referred to as different packets over multi-channel with multi-copy retransmission of lost packets scheme (DM-M). It is similar to MC-MARQ in Chapter 4.

3) Different Packets over Multi-Channel with Network Coding (DM-C)

In this scheme, before packets are transmitted, the original packets are coded with a linear random code vector as in [39]. Such coded packets are then transmitted over different available channels. The source node will keep sending new coded packets until it receives the batch ACK message from the destination node. When the destination node receives enough independent coded packets to decode the batch, an ACK message is sent to the source node. This scheme is referred to as different packets over multi-channel with network coding scheme (DM-C). It is similar to RLC-MCT in Chapter 4.

5.2.3 Two-Hop Multi-Channel Data Transmission Schemes

1) SM combined with opportunistic routing (OR_SM)

This is the same as the batch transmission scheme in the multi-channel spectrum aware opportunistic routing (MSAOR) algorithm in [85]. The algorithm employs network coding without a coordination requirement for opportunistic routing to reduce duplicated packets. The work in [85] focused on link delay analysis in the granularity of a packet, while here we will provide an insight to batch transmission capability.

2) DM-M combined with traditional routing strategy (TR_DM-M)

DM-M exploits multiple channels more efficiently compared to the SM scheme. In this scheme ACK messages are generated for each packet. This makes it be suitable for traditional routing transmission. Packets in a batch are transmitted and ACKed per hop. The source node S transmits different packets to forwarding node R over multiple available channels in each slot. The packets received by R will be ACKed to S and then forwarded to the destination node D.

3) DM-C combined with opportunistic routing (OR_DM-C)

In this scheme, data transmission explores the benefits from both network coding and opportunistic routing. Before transmission, the source node S randomly codes original packets into new coded packets. A coding vector, which indicates the coefficients of the combination of the original packets, is attached with each coded packet. Then node S broadcasts different coded packets over different multiple channels in each time slot. The number of coded packets broadcast in a slot is the same as the number of channels that S can access, which depends not only on channel availability due to primary users' activities but also on channel access contention between the source node S and the relay node R.

When the relay node R receives a packet, it checks if it is an independent packet. Node R reads the code vector of the new received packet and compares it to those of the packets received previously. If the packet is independent to the received packets in the buffer, node R saves the packet in its buffer. The independency means that the new received packet carries new packet information of the batch and it is effective and helpful for batch transmission. If the received packet is not an independent one, it will be discarded since the packets stored in the buffer of node contain the packet information of this newly received packet. To make the packet forwarding more efficient, the relay node only attempts to transmit the same number of new coded packets as the number of the independent packets received. The new coded packets transmitted are random combinations of the packets in the buffer of the relay node. When the relay node has received enough packets (N packets), it will try to keep coding packets and broadcasting the coded packets using all the available channels not used by the primary users since transmission success probability from R to D is usually higher than that from S to D.

When the destination node D receives a packet, from either S or R, it checks the independency of the packet in the same way as the relay node does and only independent packets are saved in the buffer. Also node D needs to check if it has received enough independent packets to decode the entire batch. If node D has enough packets in the buffer, it sends an ACK message to the source node and decodes the received coded packets to obtain the original packets of the batch.

Once the source node receives the batch ACK message from the destination node, it will stop transmission of the packets in this batch and it will move to transmit the next batch.

Usually the size of an ACK message is very small and the loss probability is negligible. To make the transmission of ACK more robust and faster, same copies of an ACK message can be transmitted on all the available channels. Further, in a more conservative manner, the destination node can keep sending ACK messages for a batch until no packets in the same batch are received in a specific time interval.

By using opportunistic routing, packet overhearing from S to D helps to accelerate batch transmission and improve the throughput. By using network coding, global schedulers as in [38] are avoided, which makes CRNs work in a more distributed way. The OR_DM-C is summarized as Algorithm 5.1 in Figure 5.2.

Algorithm 5.1 OR_DM-C

```
Source checks the number of accessible channels, m
Source codes m packets and broadcast them over m channels
Relay checks independent packets, codes and broadcasts new packets
If relay gets enough independent packets in buffer
        Relay keeps broadcasting packets on all available channels
Destination checks and saves independent packets
Destination sends ACK when it receives enough packets
Source moves to next batch when it receives batch ACK
```

Figure 5.2: Algorithm 5.1 OR_DM-C

5.3 Batch Delay Analysis for Different Transmission Schemes

In the previous section, we proposed to combine the DM-C single hop transmission scheme with an opportunistic routing strategy for 2-hop batch transmission in multichannel CRNs, namely OR_DM-C. We also introduced another two transmission schemes, OR_SM and TR_DM-M, for comparison. In this section, we provide some insights to these three different transmission schemes based on theoretical analysis of batch delay.

5.3.1 Batch Delay for OR_SM

In [85], the author modelled the multi-channel opportunistic CR link as an M/Geo/1 queue with the granularity of packet transmission. Packet based link delay was then derived based on the M/Geo/1 queue model. However, when combined with opportunistic routing, data transmission is batch based and no packets can be processed until the entire batch (N packets) is received. Thus batch transmission delay can provide a better insight to the batch transmission capability in multi-channel CRNs. Here we analyse OR_SM in terms of batch (transmission) delay to examine the performance of this scheme theoretically.

Before deriving the batch delay associated with OR_SM, we would like to first have a re-check of the process of batch based data transmission. In a 2-hop CRN, when data is transmitted with opportunistic routing (OR_SM, OR_DM-C) schemes, packets received at the destination node D can be divided into two groups. One group of the packets are received from the relay node R, and the other group of the packets are overheard from the source node S directly. Accordingly, the batch transmission process can be divided into two stages for analysis simplicity: In the first stage, the source node S sends the entire batch to the set of nodes R and D, {R, D}; in the second stage, the relay node R sends packets to the destination node D to decode the batch and complete the batch transmission.

For OR_SM, the batch delay can be derived according to the two stage model of batch transmission as follows.

Stage I: batch transmission from the source node S to node set {R, D}

In this stage, the source node S codes a batch of original packets into one coded packet and broadcasts it on all the k available channels in each slot. For each coded packet, it could be received by the relay node R via the link between S and R, $link_{SR}$, by the destination node D via the link between S and D, $link_{SD}$, or by both R and D. The two links, $link_{SR}$ and $link_{SD}$, compose a virtual link for packet transmission from node S to the node set {R, D}. A packet transmitted from node S to the node set {S, D} will be lost only when the transmissions over both links experience failures.

Therefore, the transmission success probability for a packet from S to the node set {R, D} can be obtained as

$$\delta_{S\{R,D\}} = 1 - (1 - \delta_{SR})^{(K\theta)}(1 - \delta_{SD})^{(K\theta)} \tag{5.1}$$

where, we assume the transmission success probabilities between nodes S and R over all the channels are same, denoted as δ_{SR}, and the transmission success probabilities between S and D over all the channels are the same, denoted as δ_{SD} .

Considering that there are totally N packets in a batch and the packet transmission success probability over the virtual link between node S and node set {R, D} can be calculated by equation (5.1), we can get the transmission delay (in the number of slots or assuming $T = 1$) of a batch from S to {R, D} in the first stage as

$$\begin{aligned} D_{S,I} &= \frac{N}{\delta_{S\{R,D\}}} \\ &= \frac{N}{1 - (1 - \delta_{SR})^{(K\theta)}(1 - \delta_{SD})^{(K\theta)}} \end{aligned} \tag{5.2}$$

Stage II: packets transmission from the relay node R to the destination node D

In the previous stage, the source node S sent N independent packets to the node set {R, D} in $D_{S,I}$. In the same period, the packet transmission success probability over $link_{SD}$ is $(1 - (1 - \delta_{SD})^{(K\theta)})$ and thus node D overheard $(1 - (1 - \delta_{SD})^{(K\theta)})D_{S,I}$ packets from node S. In stage II node D only needs $N - (1 - (1 - \delta_{SD})^{(K\theta)})D_{S,I}$ more packets from R to decode the batch. Considering the packet transmission success probability over $link_{RD}$ is $(1 - (1 - \delta_{RD})^{(K\theta)})$, as a consequence the transmission delay in the second stage is

$$
\begin{aligned}
D_{S,II} &= \frac{N - (1 - (1 - \delta_{SD})^{(K\theta)})D_{S,I}}{1 - (1 - \delta_{RD})^{(K\theta)}} \\
&= \frac{N(1 + (1 - \delta_{SR})^{(K\theta)})(1 - \delta_{SD})^{(K\theta)}}{(1 - (1 - \delta_{RD})^{(K\theta)})(1 - (1 - \delta_{SR})^{(K\theta)}(1 - \delta_{SD})^{(K\theta)})}
\end{aligned}
\tag{5.3}
$$

Combining the transmission delays in the two stages with equations (5.2) and (5.3), we can finally get the expression of the total batch delay for OR_SM as

$$
\begin{aligned}
D_S &= D_{S,I} + D_{S,II} \\
&= \frac{N((1 + (1 - \delta_{SR})^{(K\theta)})(1 - \delta_{SD})^{(K\theta)} + 1 - (1 - \delta_{RD})^{(K\theta)})}{(1 - (1 - \delta_{RD})^{(K\theta)})(1 - (1 - \delta_{SR})^{(K\theta)}(1 - \delta_{SD})^{(K\theta)})}
\end{aligned}
\tag{5.4}
$$

5.3.2 Batch Delay for TR_DM-M

In TR_DM-M, with the traditional routing strategy, packets from the source node S first are sent to the relay node R, and then these packets are forwarded to the destination node D. Without coding as in the opportunistic routing schemes, every packet needs to be acknowledged. The batch transmission delay for this scheme can be derived as follows.

Stage I: N packets from node S to node R

In this stage, the source node S sends different original packets over different multiple available channels. The packet transmission success probability from node S to node R over each channel is δ_{SR}, and there are $k = K\theta$ available channels. The capacity of the link between nodes S and R over multiple available channels is $k * \delta_{SR}$. As a result, the transmission delay for the N packets from node S to node R is then (approximately [1]) given by

$$
D_{M,I} = \frac{N}{(K\theta) * \delta_{SR}} \tag{5.5}
$$

Stage II: N packets from node R to node D

[1]For 2 packets transmitted from S to D in one hop, with each channel having a transmission success probability a, the transmission time is not 2/(2*a) slots, but $[1 + \frac{2a(1-a)}{1-(1-a)^2)}]/[1 - (1 - a)^2]$ slots instead.

In a similar way to that in stage I, we can further get the transmission delay in the second stage as

$$D_{M,II} = \frac{N}{(K\theta) * \delta_{RD}} \tag{5.6}$$

Therefore, by summating the first stage transmission delay in equation (5.5) and the second stage transmission delay in equation (5.6), the total batch delay for the TR_DM-M scheme can be obtained as

$$\begin{aligned} D_M &= D_{M,I} + D_{M,II} \\ &= \frac{N}{(K\theta) * \delta_{SR}} + \frac{N}{(K\theta) * \delta_{RD}} \end{aligned} \tag{5.7}$$

5.3.3 Batch Delay for OR_DM-C

Similar to the batch transmission with OR_SM, N independent coded packets of a batch are transmitted from the source node S to the node set {R, D} in the first stage and the relay node R then transmits the remained packets needed by the destination D in the second stage. However, in OR_DM-C, different coded packets, instead of the same one packet, are transmitted over different available channels. Accordingly, the batch transmission delay can be derived based on the two stage batch transmission model as follows.

Stage I: batch transmission from the source node S to the node set {R, D}

In this stage, a coded packet can be received from the source node via $link_{SR}$ or $link_{SD}$ over a certain available channel. Given that the packet transmission success probability from node S to node R via $link_{SR}$ over an available channel is δ_{SR} and that from node S to node D via $link_{SD}$ over an available channel is δ_{SD} , the transmission success probability from node S to node set {R, D} over a given available channel is

$$\delta_{S\{R,D\}} = 1 - (1 - \delta_{SR})(1 - \delta_{SD}) \tag{5.8}$$

Since k different coded packets are transmitted over k available channels in a time slot, the transmission delay for the batch from node S to node set {R, D} in the first stage can be obtained as

$$D_{C,I} = \frac{N}{K\theta * [1 - (1 - \delta_{SR})(1 - \delta_{SD})]} \tag{5.9}$$

Stage II: packets transmission from the relay node R to the destination node D

During stage I, the number of packets received at node D is given as

$$n_{SD} = (K\theta) * \delta_{SD} * D_{C,I} \tag{5.10}$$

In addition to these overheard packets from node S directly, the number of packets still needed from node R in this stage for node D to decode the batch and complete the batch transmission is given as

$$n_{RD} = N - n_{SD} = \frac{N\delta_{SR}(1 - \delta_{SD}))}{1 - (1 - \delta_{SR})(1 - \delta_{SD})} \tag{5.11}$$

The capacity of the link between node R and node D over multiple available channels can be calculated as $k * \delta_{SR}$. Consequently, the transmission delay for these packets over $link_{RD}$ is

$$\begin{aligned} D_{C,II} &= \frac{n_{RD}}{K\theta\delta_{RD}} \\ &= \frac{N\delta_{SR}(1 - \delta_{SD})}{K\theta\delta_{RD}(1 - (1 - \delta_{SR})(1 - \delta_{SD}))} \end{aligned} \tag{5.12}$$

Finally, with the transmission delays in the two stages as in equations (5.9) and (5.12), we can obtain the total batch delay for OR_DM-C as

$$\begin{aligned} D_C &= D_{C,I} + D_{C,II} \\ &= \frac{N}{K\theta * [1 - (1 - \delta_{SR})(1 - \delta_{SD})]} + \frac{N\delta_{SR}(1 - \delta_{SD})}{K\theta\delta_{RD}(1 - (1 - \delta_{SR})(1 - \delta_{SD}))} \\ &= \frac{N(\delta_{RD} + \delta_{SR} - \delta_{SR}\delta_{SD})}{K\theta\delta_{RD}(1 - (1 - \delta_{SR})(1 - \delta_{SD}))} \end{aligned} \tag{5.13}$$

By now, we have derived the batch delay associated with the different batch transmission schemes, equation (5.4) for OR_SM, equation (5.7) for TR_DM-M, and equation (5.13) for OR_DM-C. In the next subsection, we will examine a special case for OR_SM and OR_DM-C schemes.

5.3.4 The Special Case for OR_SM and OR_DM-C

In a 3-node CRN as in Figure 5.1, there are channel contentions between the source node S and the relay node R for packet transmission. Only when the channel conditions between node R and node D are better than those between node S and node D ($\delta_{RD} > \delta_{SD}$), packet forwarding via node R benefits network performance. This is a general case most of the time and it is an assumption in the previous batch delay analysis for OR_SM and OR_DM-C.

However, there is a special case that the channel conditions between node S and node D are better than those between node R and node D ($\delta_{RD} \leq \delta_{SD}$). In this case, all the packets are transmitted from node S to node D directly. We provide batch delay analysis for this special case for a comprehensive understanding to these transmission schemes, though it usually does not happen.

For OR_SM in the special case ($\delta_{RD} \leq \delta_{SD}$), the batch delay over the direct link between node S and node D is given as

$$D_S = \frac{N}{1 - (1 - \delta_{SD})^{K\theta}} \tag{5.14}$$

For OR_DM-C in the special case ($\delta_{RD} \leq \delta_{SD}$), the batch delay over the direct link between node S and node D is given as

$$D_C = \frac{N}{K\theta\delta_{SD}} \tag{5.15}$$

Therefore, in a general case, the batch delay for OR_SM and OR_DM-C can be calculated via equations (5.4) and (5.13), respectively. In the special case, the batch delay can be calculated via equations (5.14) and (5.15), respectively.

5.4 Performance Evaluation

In this section, we evaluate the performance of the proposed 2-hop transmission scheme, OR_DM-C. As a comparison, we also evaluate the other two transmission schemes, OR_SM and TR_DM-M. The evaluation conducted is to provide an insight to the performance of these transmission schemes.

In the evaluation, we focus on a 2-hop multi-channel CRN with lossy channels as the 3-node network topology in Figure 5.1. We first verify the theoretical analysis of batch delay associated with the three transmission schemes in Section 5.3 via simulations with Matlab programming. Then we investigate the batch delay as a function of packet loss probability with different channel availability.

In all these evaluations, the batch size is set to be $N = 8$ packets and the number of potential channels in the system is $K = 4$.

5.4.1 Theoretical Analysis Verification via Simulation

To verify the batch delay analysis in the last section, we simulate the schemes in both general case and special case. The channel availability is set to be $\theta = 0.8$ in the simulation.

1) General Case

In the first simulation, all the packet loss probabilities of the three links over a channel in the network are changed for different simulation settings, but the relationship between them is fixed as $p = 1 - \delta_{SR} = 1 - \delta_{RD} = 1 - 2\delta_{SD}$. In this scenario, the channel condition of the link between node R and node D is better than that of the direct link between node S and node D as in general in terms of packet transmission success probability ($\delta_{RD} = 2\delta_{SD}$).

In Figure 5.3, the batch delay as a function of packet loss probability is plotted for com-

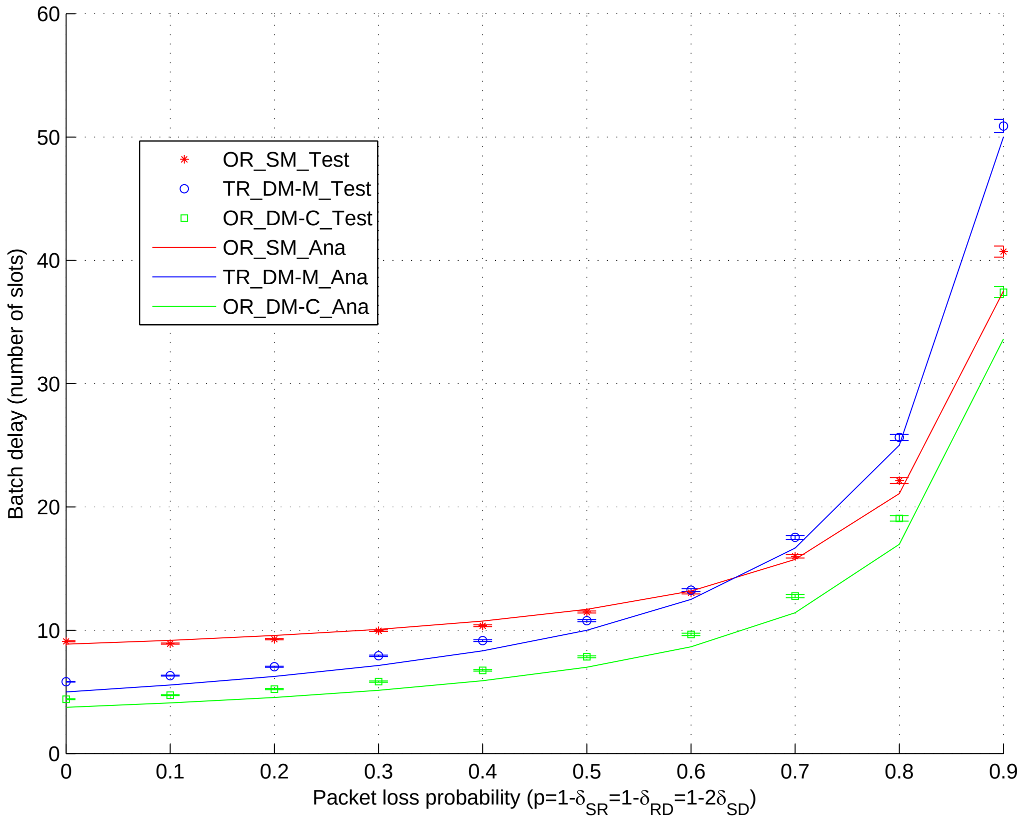


Figure 5.3: Batch delay as a function of packet loss probability: general case

parison with both simulated (test) results and analytical results for all the three schemes in different colours. For the simulated results, they are plotted with 95% confidence intervals.

It can be seen that the simulated results and the analytical results basically match though there are small gaps between them. Regarding the simplification of batch transmission process in the batch delay analysis, such small gaps are acceptable. Hence we can conclude that the analytical results are validated for the performance evaluation of batch delay in 2-hop CRNs.

2) Special Case

In the second simulation, we vary the packet loss probability of the link over a channel between node R and node D, $p = 1 - \delta_{RD}$. But at the same time we keep the packet transmission success probabilities over the other two links as constants, $\delta_{SR} = 0.8$ and $\delta_{RD} = 0.4$. Also, channel availability is set as $\theta = 0.8$. The special case comes when $p \geq 0.6$ (i.e., $\delta_{RD} \leq \delta_{SD} = 0.4$). The analytical results and the simulation results are plotted for this case in Figure 5.4.

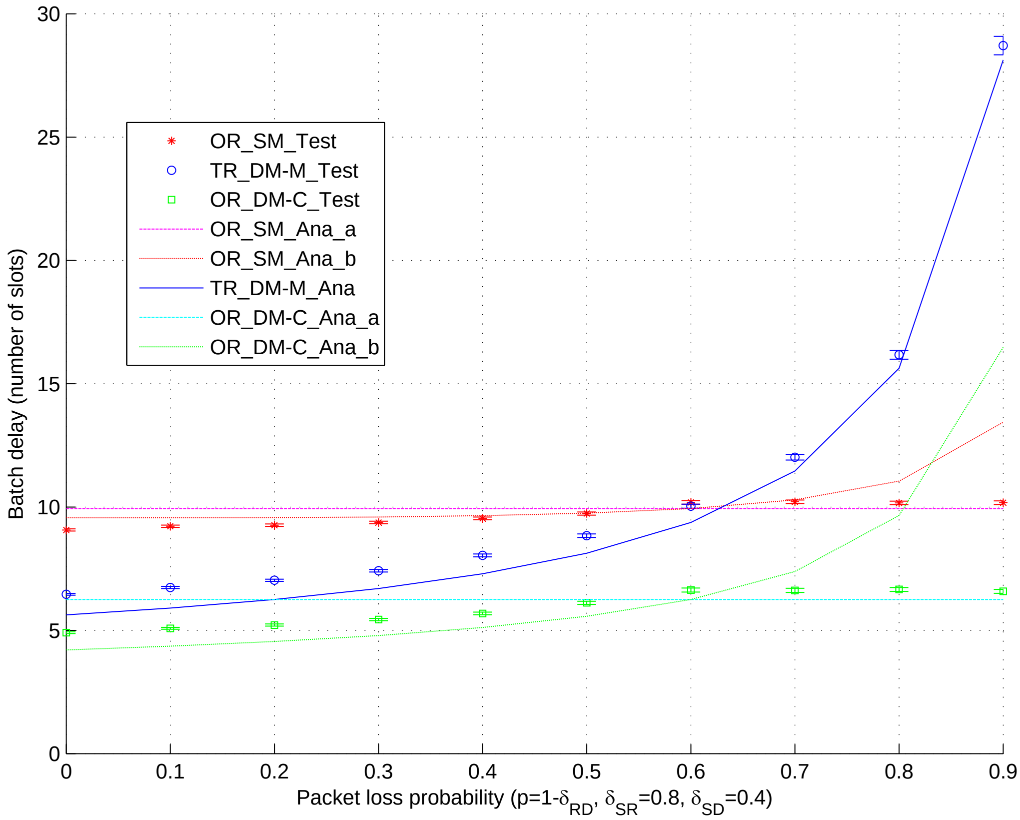


Figure 5.4: Batch delay as a function of packet loss probability: special case

In Figure 5.4, OR_SM_Ana_a, OR_SM_Ana_b, OR_DM-C_Ana_a and OR_DM-C_Ana_b are plotted with equations (5.14), (5.4), (5.15) and (5.13) respectively. OR_SM_Ana_a and OR_DM-C_Ana_a are visualized analytical results for the OR_SM scheme and the

OR_DM-C scheme in special case, while OR_SM_Ana_b and OR_DM-C_Ana_b are visualized analytical results for the OR_SM scheme and the OR_DM-C scheme in general case. It is shown that, for OR_DM-C, the simulated results match with OR_DM-C_Ana_b when $p < 0.6$ as a general case, while the results match with OR_DM-C_Ana_a when the special case appears ($p \geq 0.6$). In addition, the situation for OR_SM scheme is similar.

In a summary, the observation from Figure 5.4 validates our analytical results for the batch delay of different batch transmission schemes in the special case.

The analysis is for a deeper insight to the batch delay performance of these transmission schemes though usually the special case does not happen.

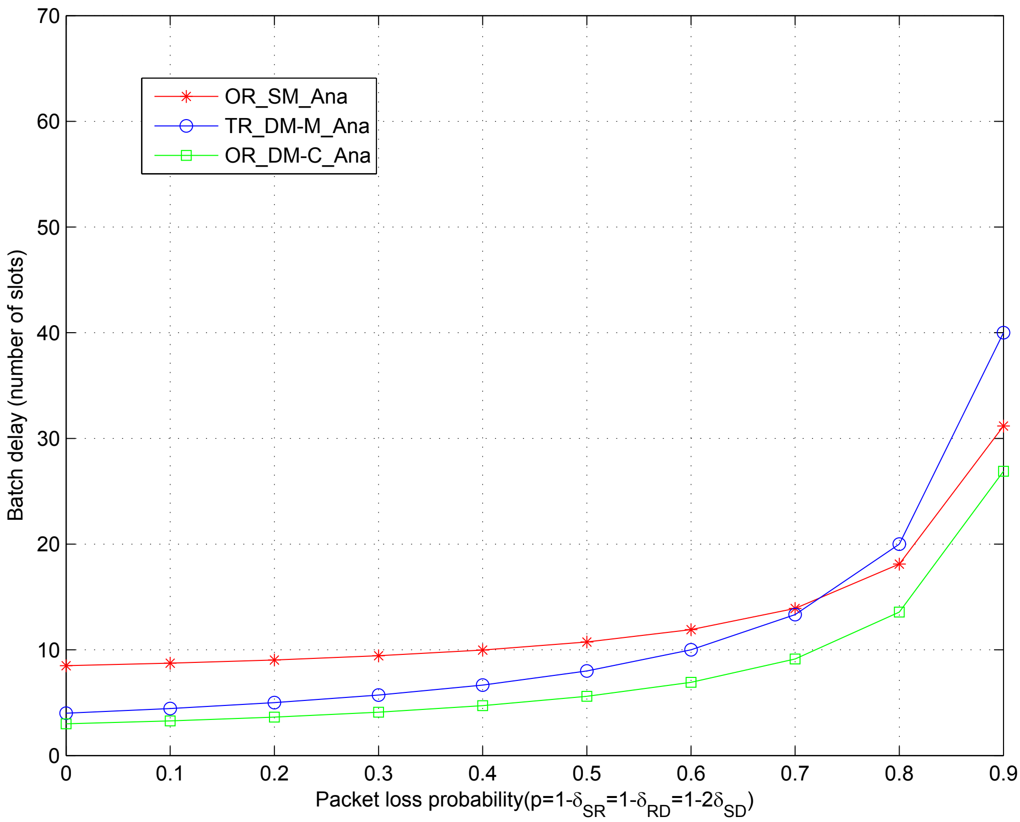


Figure 5.5: Batch delay as a function of packet loss probability with different probabilities of channel availability: $\theta = 1$

5.4.2 Performance Comparison with Analytical Results

The analytical results for the batch delay associated with the three batch transmission schemes have been validated through simulation results in the last subsection. In this subsection, performance comparison among the different transmission schemes is conducted with the visualized analytical results.

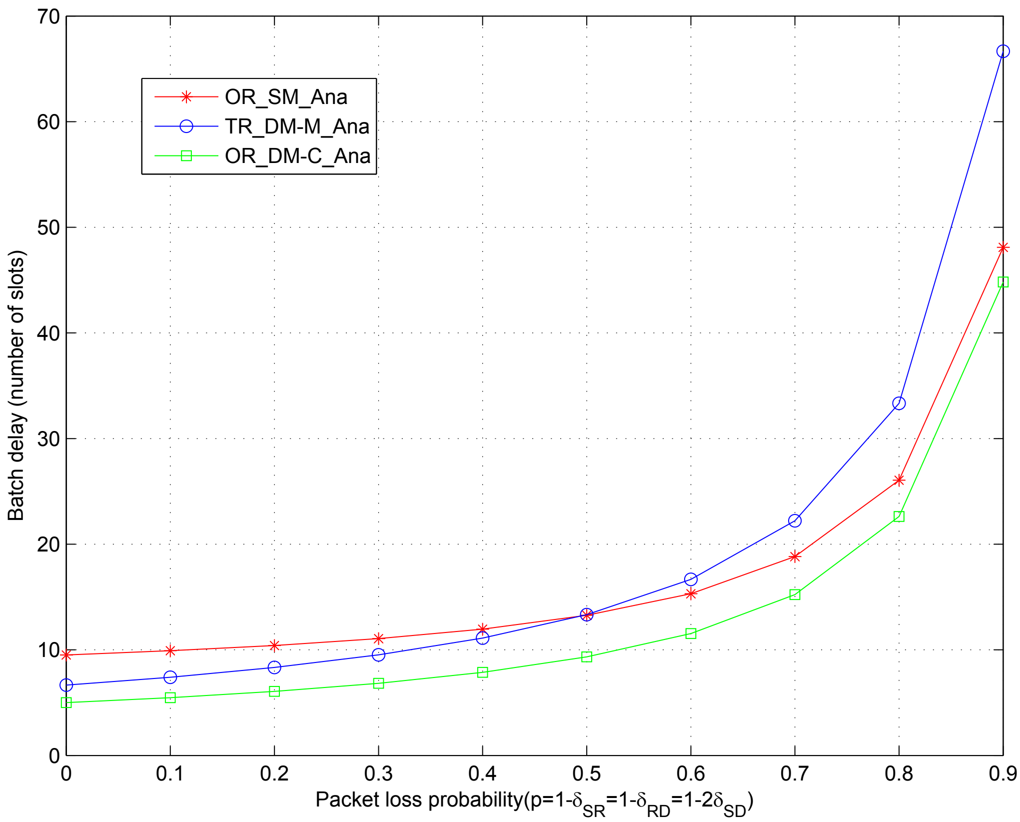


Figure 5.6: Batch delay as a function of packet loss probability with different probabilities of channel availability: $\theta = 0.6$

In Figure 5.5, Figure 5.6 and Figure 5.7, the analytical batch delay results with different channel availabilities ($\theta = 1, 0.6, 0.3$) are provided

Figure 5.5 shows that the batch delay associated with the proposed scheme, OR_DM-C, is significantly better than that of both the other schemes, for all different packet loss

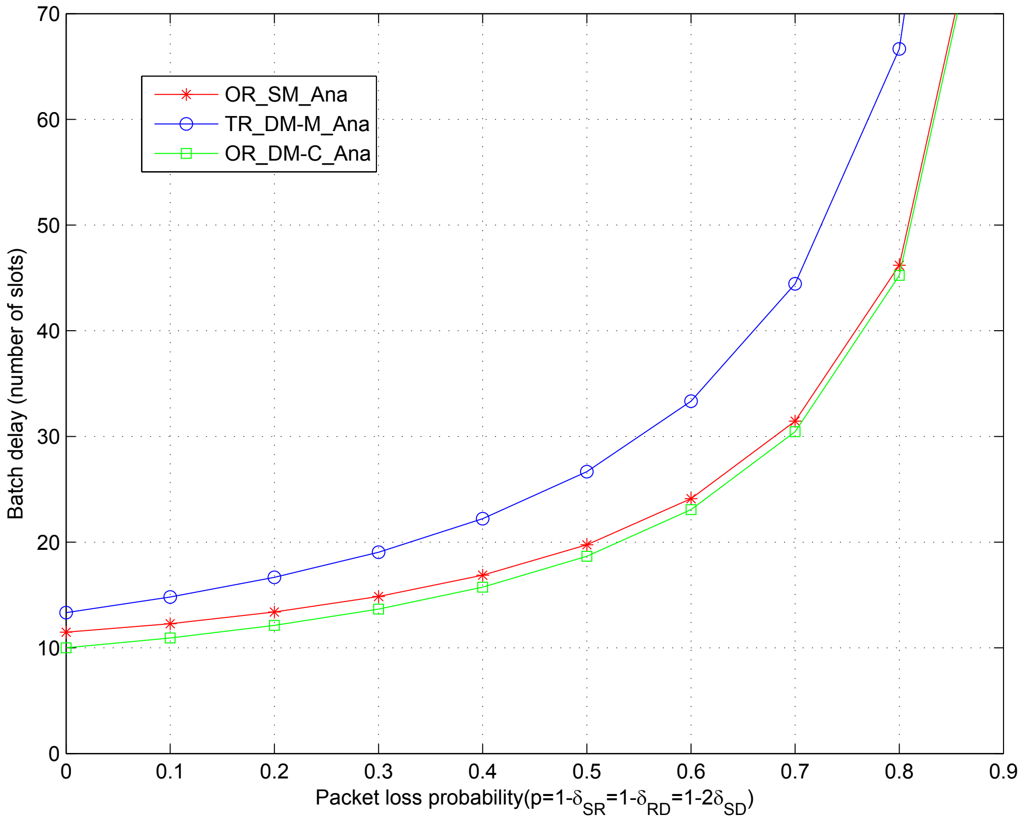


Figure 5.7: Batch delay as a function of packet loss probability with different probabilities of channel availability: $\theta = 0.3$

probabilities. Compared to OR_SM, OR_DM-C reduces the batch delay by about 60% at $p = 0.2$. The gap between these two schemes decreases as packet loss probability increases. Compared to TR_DM-M, the gap becomes larger when channel conditions are worse.

Figure 5.6 and Figure 5.7, demonstrate that when channel availability becomes lower ($\theta = 0.6$ and $\theta = 0.3$), the performance gap between OR_DM-C and OR_SM shrinks, while the gap between OR_DM-C and TR_DM-M enlarges. The shrinking of the performance gap between OR_DM-C and OR_SM is due to the fact that OR_DM-C can transmit more packets in a slot by reducing packet duplications over multiple channels in OR_SM but there are fewer packet duplications when the number of available channels decreases. The gap between OR_DM-C and TR_DM-M enlarges because packet overhearing plays a more important role as the number of available channels becomes fewer when channel availability becomes lower. Moreover, OR_DM-C outperforms both OR_SM and TR_DM-M in all the cases with different channel availability.

As shown in Figures 5.5-5.7, the batch delay for all the schemes increases when channel availability becomes lower.

5.5 Summary

In this chapter, we proposed an efficient batch transmission scheme, OR_DM-C, for multi-channel CRNs based on the combination of network coding and an opportunistic routing strategy. To provide a good insight into data transmission capability of the proposed scheme, we conducted theoretical analysis of the batch delay and then verified it via simulation. For comparison, other two schemes are also analysed. As indicated in both the analytical results and the simulation results, OR_DM-C can reduce batch transmission delay significantly and it outperforms the other two schemes in various scenarios (different packet loss probability and channel availability).

In Chapter 3 to Chapter 5, we focused on routing and transmission design in multi-radio CRNs. In next chapter we will deal with channel assignment and routing design in single transceiver CRNs.

Chapter 6

Maximum Flow-Segment Based Channel Assignment and Routing in Cognitive Radio Networks

6.1 Overview

In previous chapters, we designed an opportunistic routing algorithm for multi-channel cognitive radio networks. We focused on the analysis and improvement of efficient batch transmission schemes over multiple channels, which is a key component of the opportunistic routing algorithm. For those works, we assumed that each cognitive radio user is equipped with multiple transceivers. In practice, there are situations, e.g. wireless sensor networks, where each CR user is equipped with only one transceiver for low cost and easy implementation. In such cases, channel switching is required for CR users to communicate in a dynamic spectrum access environment. In a multi-hop CRN, frequent channel switching may result in a dramatic increase in end-to-end delay when a traffic flow switches between a number of channels along its path. Therefore, it is important to investigate the impacts of channel switching in the channel assignment and routing design in CRNs.

In this chapter, we propose a new maximum flow-segment (MFS) based channel assignment and routing scheme for single transceiver CR nodes in a multichannel, multi-hop CRNs. Our MFS based scheme minimizes the number of channel switches by selecting the available channels that connect the maximum number of nodes in a routing path, which therefore provides a reduction of the end-to-end delay. In addition, we derive an efficient channel information dissemination algorithm that integrates the MFS based channel assignment scheme into the AODV on-demand routing protocol [50].

Extensive simulations and analysis show that our MFS based scheme achieves more than 50% reduction in the end-to-end delay and more than 10% improvement in the peak throughput as compared to the link based scheme. We demonstrate that our MFS based scheme maintains a higher and more stable throughput under heavy load, and it scales well with network size. In the more complicated network topologies with intersecting nodes, we show that our MFS approach achieves almost a 50% reduction in the end-to-end delay.

The remainder of this chapter is organized as follows. Section 6.2 introduces the system model and a motivating case study. In Section 6.3, we present the MFS based algorithm and its integration into an on-demand routing protocol in detail. Simulation results and performance evaluations are discussed in Section 6.4. Finally we summarize this chapter in Section 6.5.

6.2 System Model and Motivations

6.2.1 System Model

We focus on a single transceiver, multi-hop CRN. In the network, we assume that there are totally K frequency channels in the available channel pool, which is denoted as $A = \{ch_i | i = 1, ..., K\}$. A CR node is assumed to have adequate spectrum sensing capability that can provide the network layer with the available channel information. The available channel set for node n is denoted as A_n ($A_n \subseteq A$). Each CR node is equipped with a single half-duplex CR transceiver, which can switch to and operate on different channels but can only receive or transmit packets on a selected channel at a given time.

6.2.2 A Motivating Case Study

Consider the following scenario in Figure 6.1. There are seven CR nodes in a chain topology network. The available channel sets at these seven nodes are A_1, A_2, ..., A_7, respectively. There is a traffic flow from node 1 to node 7.

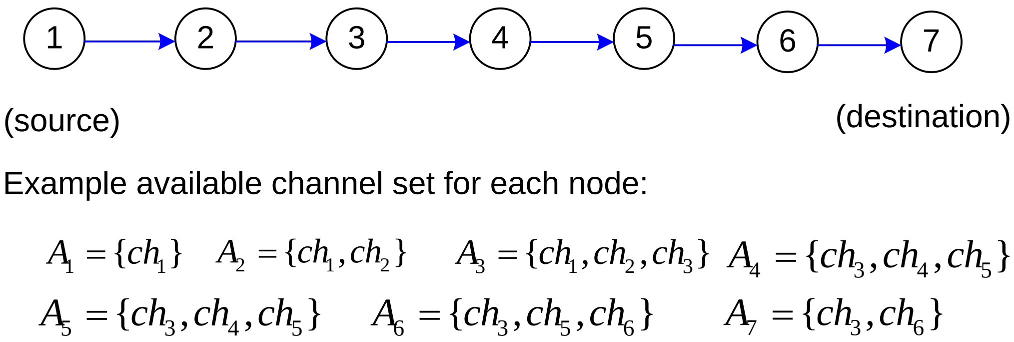


Figure 6.1: An example flow in a cognitive radio network

In a traditional flow based channel assignment algorithm [7, 8], a common channel usually is selected for all the nodes along the flow to avoid channel switching. However, since different users have different available channel sets in a CRN, we can see that there is no common channel that can connect all the seven nodes in the network, which means that the traditional flow based channel assignment algorithm is infeasible.

As an alternative way, a link based channel assignment [8] can be employed to select an operating channel for a flow in CRNs. However it may suffer from performance degradation due to channel switching. With a link based channel assignment approach, a channel is selected independently for each link based on factors such as link quality, but not based on the more sophisticated measures such as minimizing the number of channel switches for the entire flow. For example a link based assignment for the flow in Figure 6.1 could be $\{ch_1, ch_2, ch_3, ch_4, ch_5, ch_6\}$ which is not desirable since every node forces a channel switch. Under such a channel assignment scheme, there are five channel switches when a packet is transmitted from the source node 1 to the destination node 7. Considering that there is a switching delay for each channel switch, the end-to-end delay for the flow with such a channel assignment could be drastically increased by the five channel switches.

Note that there is a different channel assignment that minimizes the number of channel switches: $\{ch_1, ch_1, ch_3, ch_3, ch_3, ch_3\}$. In this channel assignment, there is only one channel switch on the path of the flow, which is at node 3 between the operating channel ch_1 for the link connecting node 2 and node 3 and the operating channel ch_3 for the link connecting node 3 and node 4. Since there is no common channel that can connect all the seven nodes along the flow without any channel switch, this channel assignment with only one channel switch is the most ideal channel assignment for the flow, considering the influence of the channel switching delay on the end-to-end delay.

In the next section, we first develop our scalable channel assignment algorithm which always finds the channel assignment with the minimum number of channel switches. Then we integrate the channel assignment algorithm into the AODV on-demand routing protocol with an efficient channel information dissemination algorithm.

6.3 MFS Based Channel Assignment and Routing Algorithm

To facilitate the algorithm design, we first define the *flow-segment* (*FS*) and the *maximum flow-segment* (*MFS*) with regards to a designated node n in a specific flow f.

FS: A *FS* (*flow-segment*) of node n in flow f is a set of successive upstream nodes (of node n) that can be connected with the same frequency channel. The nodes in a FS are ordered in the reverse direction of f.

For example, in Figure 6.1, node 7 has four FSs: {7, 6}, {7, 6, 5}, {7, 6, 5, 4} and {7, 6, 5, 4, 3} in the flow. FS {7, 6} can be connected with the frequency channel ch_3 or ch_6, FS {7, 6, 5} can be connected with the frequency channel ch_3, FS {7, 6, 5, 4} can be connected with the frequency channel ch_3, and FS {7, 6, 5, 4, 3} can be connected with the frequency channel ch_3.

MFS: For a given node in flow f, there can be multiple FSs. Among these, the FS that contains the maximum number of nodes is defined as the *MFS* (*maximum flow-segment*) of the node.

For example, in Figure 6.1, the MFS of node 7 is {7, 6, 5, 4, 3}, which contains the maximum number of nodes, five, among the four FSs of node 7. The nodes in the MFS can be connected with the frequency channel ch_3.

Our proposed new channel assignment and routing algorithm proceeds as follows:

1) Build the MFS at the destination node of a flow;

2) Go to the last node of the MFS in the previous step, and build the MFS for this node;

3) Repeat step 2);

4) End when the source node is reached.

As an example, consider the flow in Figure 6.1. To build MFSs along the routing path, we first obtain the MFS {7, 6, 5, 4, 3} at the destination node 7. Then at the last node 3 of the previous MFS, we build the MFS {3, 2, 1} for node 3, where source node 1 is reached. Thus the nodes along the flow can be divided into two flow segments with MFS {7, 6, 5, 4, 3} and MFS {3, 2, 1}. After that, ch_3 is assigned to the links between the adjacent nodes in the first MFS {7, 6, 5, 4, 3}. Similarly, ch_1 is assigned to the links between the adjacent nodes in the second MFS {3, 2, 1}. As a result, the channel assignment for the links along the flow is $\{ch_1, ch_1, ch_3, ch_3, ch_3, ch_3\}$, which is the ideal assignment we identified in the motivating example in the previous section.

6.3.1 Route Discovery and Channel Information Dissemination

In this section, we integrate our new MFS based channel assignment approach into the AODV on-demand routing protocol. Note that the same can be done for other on-demand routing protocols in a straight forward manner.

In on-demand routing protocols, when a source node wants to transmit data to a destination node, it first broadcasts a routing request (RREQ) in the network for route discovery. The RREQ is forwarded by intermediate nodes after updating their routing information. When the destination node receives the RREQ, it chooses the best route and sends a routing reply (RREP) carrying route decision back to the source node.

In CRNs, an important requirement for routing design is that a node on the route path not only needs to know who the next hop node to receive the packets is but also needs to know on which channel the packets should be transmitted. Therefore the available channel information at each node on the route path is carried with RREQ in the route discovery process to help the destination node make routing decisions jointly with channel assignment [73, 83]. Based on the basic idea of our MFS based algorithm, we propose an efficient algorithm, *Algorithm-A*, to simplify the channel information dissemination process in the route discovery stage.

A K-tuple, $TP_n = \{(M_n^{ch_1}, flag_n^{ch_1}), ..., (M_n^{ch_K}, flag_n^{ch_K})\}$, is used to record channel information for the upstream nodes of node n. The symbol $M_n^{ch_i}(1 \leq i \leq K)$ denotes the number of successive upstream nodes from node n, including n, which can be connected by ch_i on the routing path of a flow. The symbol $flag_n^{ch_i}$ indicates whether there is an existing flow operating on ch_i through one of the upstream $M_n^{ch_i}$ nodes of node n. The K-tuple TP_n is added into RREQ to carry the available channel information of node n.

When a source node (node 1) tries to discover a route to a destination node (node N), the source node first sets the value of the K-tuple, TP_1, according to its available channel information. For all the channels in the channel pool $A = \{ch_i | i = 1, ..., K\}$, the source node checks if ch_i is available or not first. If ch_i is available for node 1, it sets $M_1^{ch_i}$ to be 1, otherwise sets $M_1^{ch_i}$ to be 0. If there is an existing flow passing through node 1 and operating on ch_i, $flag_1^{ch_i}$ is set to be *true*; otherwise $flag_1^{ch_i}$ is set to be *false*. Then, node 1 broadcasts the RREQ with the available channel information to its neighbors to discover routes.

When an intermediate node n receives the RREQ, it updates the RREQ according to its channel information. If ch_i is available for node n, the number of successive upstream nodes is increased by one, i.e. $M_n^{ch_i} = M_{n-1}^{ch_i} + 1$. If there is an existing flow passing through node n and operating on ch_i, $flag_n^{ch_i}$ is set to be *true*; otherwise it keeps the original value, i.e. $flag_n^{ch_i} = flag_{n-1}^{ch_i}$. If ch_i is unavailable at the intermediate node n, $M_n^{ch_i}$ is reset to be 0 and $flag_n^{ch_i}$ is reset to be *false*. After calculating TP_n, node n updates and forwards the RREQ accordingly. The pseudo code of *Algorithm-A* is shown in Figure 6.2.

We use the example shown in Figure 6.1 to illustrate the proposed *Algorithm-A*. When node 1 sets up a route to node 7, it finds that only ch_1 is available and there is no existing flow passing through node 1. Node 1 sets $M_1^{ch_1}$ and $flag_1^{ch_i}$ $(i = 1, ..., 6)$ to be 1 and 0 (*false*), respectively. Since all the other channels, ch_2, ch_3, ..., ch_6, are not available,

Algorithm A. Channel information dissemination

begin

node n sets TP_n

for i=1, …, K

$M_0^{ch_i}=0$, $flag_0^{ch_i}=false$ //initialization

if $ch_i \in A_n$ //available frequency channel for node n

$M_n^{ch_i}=M_{n-1}^{ch_i}+1$

if $ch_i \in A_n^{'}$ //an operating channel of existing flow

$flag_n^{ch_i}=true$

else $flag_n^{ch_i}=flag_{n-1}^{ch_i}$

else $M_n^{ch_i}=0$, $flag_n^{ch_i}=false$

$TP_n=\{(M_n^{ch_1}, flag_n^{ch_1}),...,(M_n^{ch_K}, flag_n^{ch_K})\}$

if $n<N$

TP_n is broadcasted with RREQ

if $n=N$

Assign channels on the path with Algorithm-B

end

Figure 6.2: Algorithm-A, channel information dissemination

the value of $M_1^{ch_i}$ $(i=2,...,6)$ is set to be 0. Then, node 1 broadcasts the RREQ with $TP_1=\{(1,0),(0,0),(0,0),(0,0),(0,0),(0,0)\}$ to its next hop neighbors. When node 2 receives the RREQ from node 1, it checks its available channels first. Since ch_1 and ch_2 are available and no existing flows pass through node 2, it updates RREQ with

$TP_2 = \{(2,0),(1,0),(0,0),(0,0),(0,0),(0,0)\}$ and broadcasts it. Following the same way, we can get:

$TP_3 = \{(3,0),(2,0),(1,0),(0,0),(0,0),(0,0)\}$,

$TP_4 = \{(0,0),(0,0),(2,0),(1,0),(1,0),(0,0)\}$,

$TP_5 = \{(0,0),(0,0),(3,0),(2,0),(2,0),(0,0)\}$,

$TP_6 = \{(0,0),(0,0),(4,0),(0,0),(3,0),(1,0)\}$,

$TP_7 = \{(0,0),(0,0),(5,0),(0,0),(0,0),(2,0)\}$.

When the destination node receives RREQ, it uses the TP_n value for channel assignment, discussed in the next subsection.

6.3.2 Route Setup and Channel Assignment

When the destination node N receives RREQ, it computes the minimum hop route as specified in AODV. Then node N begins to assign channels based on its TP_N value for the new flow. The maximum $M_N^{ch_j}$ is selected from TP_N, which is the number of nodes in the MFS of node N in the new flow. Accordingly, ch_j can be selected as a common channel for all the nodes in the MFS.

Let $l_{n-1,n}$ denotes the link between nodes n-1 and n, and let $Sh_{n-1,n}$ represent the frequency channel assigned to $l_{n-1,n}$. Thus node N assigns ch_j to the link between node N and its upstream node N-1, i.e. $Sh_{N-1,N} = ch_j$. $Sh_{N-1,N}$ is piggybacked in RREP and sent to the upstream node N-1 on $Sh_{N-1,N}$. If multiple channels, denoted as A_N^* ($A_N^* \subseteq A_N$), are available for all the nodes in the MFS, ch_k ($ch_k \in A_N^*$) with $flag_N^{ch_k} = true$ is selected. Moreover, if there are multiple channels with *true* flags, denoted by$\hat{A_N}$ ($\hat{A_N} \subseteq A_N^*$), the channel can be selected randomly from $\hat{A_N}$.

When an intermediate node n receives the RREP from its downstream node n+1, it checks the channel assignment, $Sh_{n,n+1}$, for the link $l_{n,n+1}$. If $Sh_{n,n+1}$ is unavailable for its upstream node n-1, a new MFS is required at this intermediate node. We call such node

Algorithm B. Channel assignment

begin

  if *n=N* or $Sh_{n,n+1} \notin (A_n \cap A_{n-1})$ *//decision node*

    $A_n^* = \{ch_i \mid ch_i = \arg\mathrm{Max}(M_n^{ch_i}), i = 1, 2, ..., K\}$

    if $|A_n^*| > 1$

      $A_n^{\wedge} = \{ch_i \mid ch_i \in A_n^*, flag_n^{ch_i} = true\}$

      if $A_n^{\wedge} \neq \varnothing$

        $Sh_{n-1,n} = ch_i, ch_i \in A_n^{\wedge}$

      else $Sh_{n-1,n} = ch_i, ch_i \in A_n^*$

    else $Sh_{n-1,n} = ch_i, ch_i \in A_n^*$

  else if $A_n^{'} \neq \varnothing$ *//intersecting node*

    if $A_n^{'} \cap (A_n \cap A_{n-1}) \neq \varnothing$

      if $Sh_{n,n+1} \in (A_n^{'} \cap (A_n \cap A_{n-1}))$

        $Sh_{n-1,n} = Sh_{n,n+1}$

      else $Sh_{n-1,n} = \arg\mathrm{Max}(M_n^{Sh_{n,n+1}}, M_n^{ch_i}), ch_i \in A_n^{'}$

    else $Sh_{n-1,n} = Sh_{n,n+1}$

  else $Sh_{n-1,n} = Sh_{n,n+1}$ *//other nodes*

end

Figure 6.3: Algorithm-B, channel assignment

a decision node and the channel is assigned in the same way as that at the destination node N.

Now consider the situations with intersecting flows. If the new flow intersects with an existing flow at node n (called intersecting node) and $Sh_{n,n+1}$ is the operating channel of the existing flow, $Sh_{n,n+1}$ is assigned to the link $l_{n-1,n}$, i.e. $Sh_{n-1,n} = Sh_{n,n+1}$. If node n is an intersecting node but $Sh_{n,n+1}$ is not the operating channel of the existing flow, one of the channels, $Sh_{n,n+1}$ or ch_i ($ch_i \in A_n^{'}$), where $A_n^{'}$ denotes the operating channel set of the existing flows through node n), that can connect the maximum number of upstream nodes is selected.

When the node n is neither a decision node nor an intersecting node, it selects the same channel as that is used in its downstream link, i.e. $Sh_{n-1,n} = Sh_{n,n+1}$.

All nodes on the path from the destination to the source perform channel assignment according to the above procedure. Then the route from the source to the destination is built up and the channels are assigned to all the links along the path. The pseudo code of the algorithm described above is shown in Figure 6.3, which is named *Algorithm-B*.

Consider the example in Figure 6.1. Before the destination node 7 sends its RREP to node 6, it checks the value of TP_7 and finds ch_3 can connect five nodes and ch_6 can connect only two nodes. Therefore, nodes 7, 6, 5, 4, 3 consist of the MFS of node 7 and ch_3 is assigned to the link $l_{6,7}$. The channel assignment of node 7, $Sh_{6,7} = ch_3$, is piggybacked in the RREP to node 6. Since nodes 6, 5, 4 are neither decision nodes nor intersecting nodes, they select the same channel as that used by their downstream links, i.e. $Sh_{3,4} = Sh_{4,5} = Sh_{5,6} = Sh_{6,7} = ch_3$. Node 3 is the last node in the MFS of node 7 and a new MFS is required to be built at this node. Thus node 3, as a decision node, selects frequency channel ch_3 for its downstream link $l_{3,4}$and ch_1 for the upstream link since ch_1 can connect all the nodes within the MFS of node 3 ($\{3, 2, 1\}$). Node 2 selects the same channel as that used by its downstream link $l_{2,3}$ for its upstream link $l_{1,2}$, i.e. $Sh_{1,2} = Sh_{2,3} = ch_1$, because it is also neither a decision node nor a intersecting node. When node 1 receives the RREP from node 2, it chooses ch_1 for the link $l_{1,2}$. Consequently, by now, the route is set up and the ideal channel assignment $\{ch_1, ch_1, ch_3, ch_3, ch_3, ch_3\}$ is achieved.

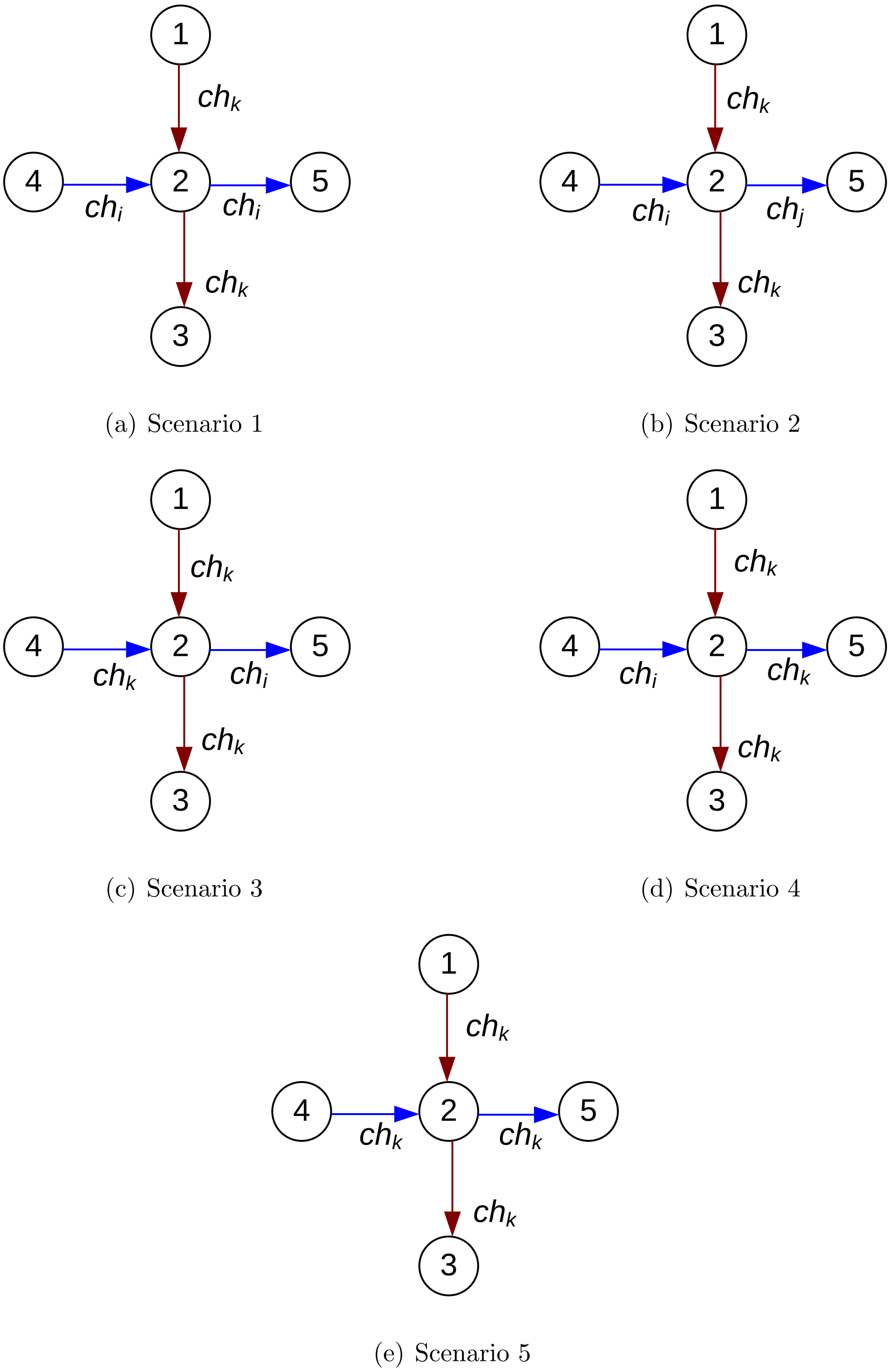
6.3.3 Channel Selection at Intersecting Nodes

In this subsection, we consider network scenarios where a new flow intersects with an existing flow at an intersecting node. A straight forward scheme is to let the new flow choose the same operating channel with the existing flow - the sharing scheme. Although the sharing scheme is able to avoid channel switching at the intersecting node, it often splits the established MFS for the new flow in order to share the channel with the existing flow. This introduces additional channel switches at the upstream node and the downstream node of the intersecting node for the new flow. With the proposed MFS based switching scheme, the intersecting node selects an operating channel according to *Algorithm-B* for the new flow. It maintains the integrity of the MFS for the new flow with only a single necessary channel switch at intersecting node.

We compare the sharing scheme with our proposed MFS based switching scheme in terms of the channel switching delay under the five different scenarios as shown in Figure 6.4. In Figure 6.4, an existing flow, operating on ch_k, passes through nodes 1, 2 and 3. The new flow transfers through nodes 4, 2, and 5. Note that Figure 6.4 is only a part of the two longer flows around intersecting node 2. The five scenarios are obtained with different channel assignment results generated from the MFS based channel assignment scheme.

1) Scenario 1 (Figure 6.4(a)): ch_i is assigned to both upstream link (left) and downstream link (right) of node 2 (intersecting node) for the new flow.

In this switching scheme scenario, node 2 is included in an MFS on the path of the new flow. Another requirement to achieve such a channel assignment is that ch_i can connect more upstream nodes of node 2 than the operating channel of the existing flow,



(a) Scenario 1 (b) Scenario 2 (c) Scenario 3 (d) Scenario 4 (e) Scenario 5

Figure 6.4: Various channel assignment scenarios under intersecting flows

ch_k. With the current channel assignment, node 2 has to switch between ch_i and ch_k. In a *channel switching cycle*, defined as the period when a node switches its operating channel for a round among all the channel assigned to the upstream and downstream links connected this node one by one and finally switches the operating channel back to the original channel, node 2 need to switch its operating channel twice, from ch_i to ch_k then back to ch_i. Assume that a node switching from one channel to another results in a delay of T. Then in scenario 1 the delay of a channel switching cycle is 2T.

If the sharing scheme is adopted, ch_k is assigned to node 2's upstream and downstream links of the new flow. In this case, there will be no channel switching delay at node 2. However channel switching will happen at the other two nodes, node 4 and 5, in the MFS since the channel assignment breaks the MFS at node 4 and node 5. The total delay will be 4T, 2T at node 4 and 2T at node 5.

The above analysis shows that the switching scheme performs better than the sharing scheme in this scenario.

2) Scenario 2 (Figure 6.4(b)): ch_i is assigned to upstream link while ch_j is assigned to downstream link of node 2.

The channel assignment in this scenario is generated from the case that ch_j is the common channel of an MFS of node 2's downstream nodes but it is not available for its upstream node, node 4, and the common channel of the MFS of node 2 is ch_i. With a similar analysis as that in Scenario 1, the channel switching delay is 4T for the sharing scheme since this scheme breaks two MFSs at node 4 and node 5. The channel switching delay is 3T for the switching scheme. Thus the switching scheme is better than the sharing scheme in this scenario.

3) Scenario 3 (Figure 6.4(c)): ch_k is assigned to upstream link while ch_i is assigned to downstream link of node 2.

The channel assignment in this scenario can be a result of the case, under the MFS based scheme, that node 2 is in the MFS of its downstream nodes connected with ch_i, but the operating channel of the existing flow, ch_k, can connect more upstream nodes of node 2 in the new flow than ch_i and the other available channels. The channel switching delay at node 2 is 2T for the switching scheme, while the sharing scheme introduces a switching delay of 2T at node 5. Therefore, the performance of both schemes is same in this scenario.

4) Scenario 4 (Figure 6.4(d)): ch_i is assigned to upstream link while ch_k is assigned to downstream link of node 2.

The channel assignment in this scenario is the result of the case that ch_k is the common channel for an MFS of the downstream nodes of node 2, but ch_k is not available for the link between node 4 and node 2, and ch_i is the common channel that can connect the maximum number of node 2's upstream nodes. The channel switching delay at node 2 is 2T for the switching scheme, while the sharing scheme is not applicable.

5) Scenario 5 (Figure 6.4(e)): ch_k is assigned to both upstream link and downstream link of node 2.

The channel assignment in this scenario can be a result of the case that the operating channel of the existing flow ch_k is the common channel of an MFS of node 2's downstream nodes, and ch_k is also available for the link between node 4 and node 2. Since under the switching scheme the assigned channel for the links connecting node 4, node 2 and node 5 in the new flow is the same channel as the operating channel of the existing flow through node 1, node 2 and node 3, there is no difference with the channel assignment under the sharing scheme. Accordingly, the performance of both schemes is same in this scenario.

Based on the above analysis, we conclude that the switching scheme performs better than the sharing scheme and the proposed MFS based channel assignment algorithm is effective in reducing a channel switching delay at the intersecting node.

6.4 Simulation and Evaluation

To evaluate the performance of our MFS based channel assignment and routing algorithm, simulations are conducted using ns-2 [105]. We have extended the ns-2 implementation to support dynamic channel switching in multi-channel single interface CRNs. Our extension is partly based on the multi-channel multi-interface module of [106, 107] and a CRN patch [108].

In the simulations, we use a modified 802.11 DCF with added support of channel switching control at the MAC layer. Constant bit rate (CBR) over UDP is used for the traffic source. In all the simulations, the packet size is set to be 1000 bytes, the distance between neighboring nodes is 200m, the transmission range is set to be 250m and the channel switch delay is set to be 10ms [6, 73, 109].

We first compare the MFS based scheme with link based scheme in a chain topology with a single flow. Then we investigate the channel selection mechanism at the intersecting node for the MFS based scheme in a two-flow network topology.

6.4.1 MFS Based Scheme vs. Link Based Scheme

In the chain topology simulation, we measure the end-to-end delays and the throughputs of the two schemes. In the simulated network, there are totally 6 feasible channels in the channel pool. The available channel set for each node is composed by randomly designating 4 out of the 6 channels. In the MFS based scheme, we assign operating channels according to the MFS algorithm described in Section 6.3. In the link based scheme, we randomly select one channel from the intersection of the two neighboring nodes available channel sets as the operating channel for the link between the nodes. We vary the length of the path from 4 hops to 15 hops. At each step, we simulate both the MFS and link based schemes for 10 times with different available channel sets.

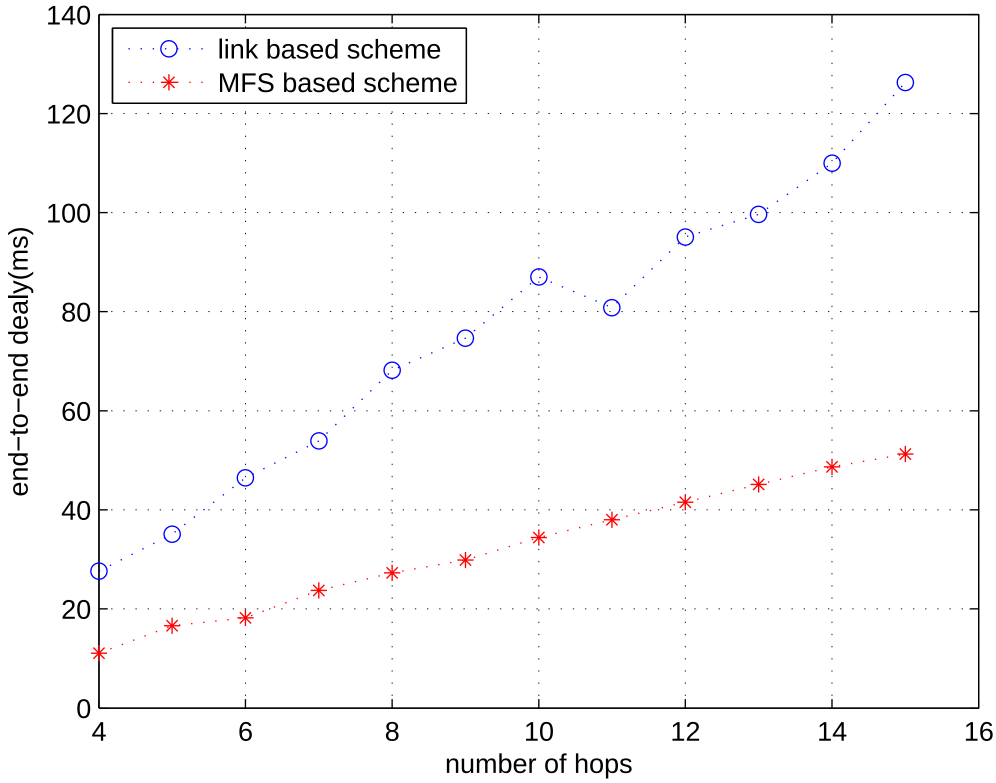


Figure 6.5: End-to-end delay of our proposed MFS based scheme and link based scheme

The results in Figure 6.5 show that our MFS based scheme reduces the end-to-end delay by more than 50% as compared to the link based scheme. For example, when the length of the path is 8 hops, the end-to-end delay for the link based scheme is approximately 68 ms, while the delay for our MFS based scheme in only around 27 ms, producing a significant reduction by about 60%. For the simulation results, the margin of error at 95% confidence ranges from 3.7ms to 8.8ms for different network sizes under the MFS based scheme, while the margin of error at the same confidence level ranges from 5.5 ms to 9.9 ms under the link based scheme. Moreover, the MFS based scheme scales better with networks sizes. This can be observed from Figure 6.5 in that as the number of the routing hops increases, the end-to-end delay of the MFS based scheme grows slower than that of the link based scheme. More specifically, the end-to-end delay under the link based scheme increases by about 9 ms with the one hop increment of the routing path, while the delay under the MFS based scheme goes up by only 3.6 ms per hop, a 60% reduction of the increasing rate of the end-to-end delay varying with the network size.

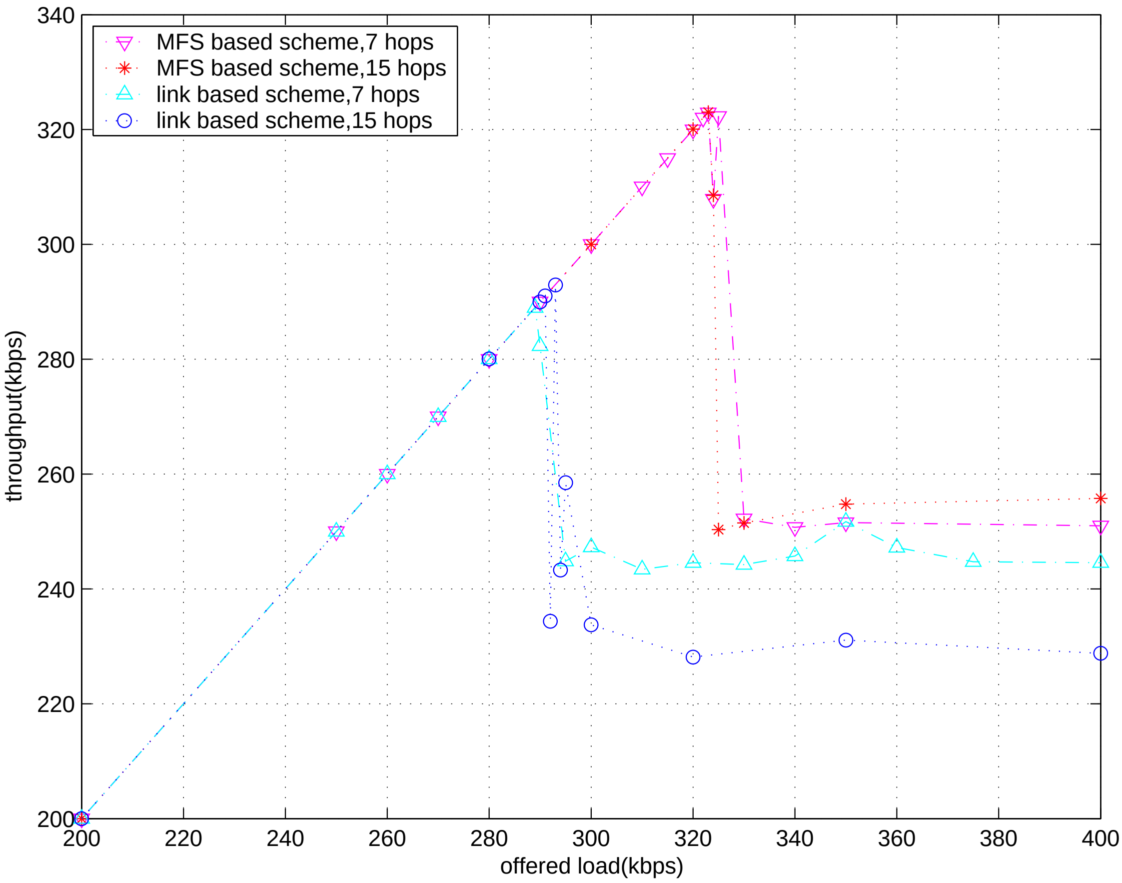


Figure 6.6: Throughput of our proposed MFS based scheme and link based scheme varying with offered load

In the next experiment, we set up two networks of 7 hops, and 15 hops respectively. The throughputs of the two networks are measured over a range of offered load. In Figure 6.6 the results show that the MFS based scheme achieves a higher maximum throughput, over 320kbps, in both 7-hop and 15-hop networks, while the link based scheme can only reach approximately 290kbps. When the network is in the overload state, the throughput of the link based scheme experiences a significant drop as the number of hops increases, while there is little variation in the saturated throughput of the MFS based scheme (our simulation results show that the saturated throughputs for the two network experiments are within the confidence interval of each other), which again demonstrates the scalability of the MFS based scheme.

6.4.2 MFS Based Switching Scheme in a Two-Flow Network

In this subsection, we evaluate the channel switching schemes at the intersecting node in a two-flow network. We compare the MFS based switching scheme and the sharing scheme as discussed in Section 6.3.3. In a two-flow network as shown in Figure 6.7, there is an existing vertical flow 0, occupying channel 5 for all the links along the flow, $\{ch_5, ch_5, ch_5, ch_5\}$. A new horizontal flow 1 is injected intersecting with the existing flow 0 at node 5.

We simulate the two channel assignment schemes at the intersecting node 5. With the MFS based switching scheme, the new flow chooses the operating channels with the maximum flow segment length $\{ch_1, ch_1, ch_3, ch_3, ch_3, ch_3\}$ for the new flow 1. In this case different channels have been chosen at the intersection node 5 for the two flows, ch_5 for flow 0 and ch_3 for flow 1. The intersecting node 5 will have to switch between ch_3 and ch_5 to serve the two flows. With the sharing scheme, the new flow 1 switches to ch_5 at node 5 to avoid channel switching at the intersecting node, i.e. $\{ch_1, ch_1, ch_3, ch_5, ch_5, ch_3\}$ for the new flow 1. However the new flow 1 has to take two more channel switches to reach the destination in the sharing scheme at the upstream node, node 4, and the downstream node, node 6, of the intersecting node.

Figure 6.8 shows the end-to-end delay of the different channel assignment schemes. The MFS based switching scheme (right hand side of Figure 6.8) achieves almost a 50% reduction in the end-to-end delay compared to the sharing scheme (left hand side of

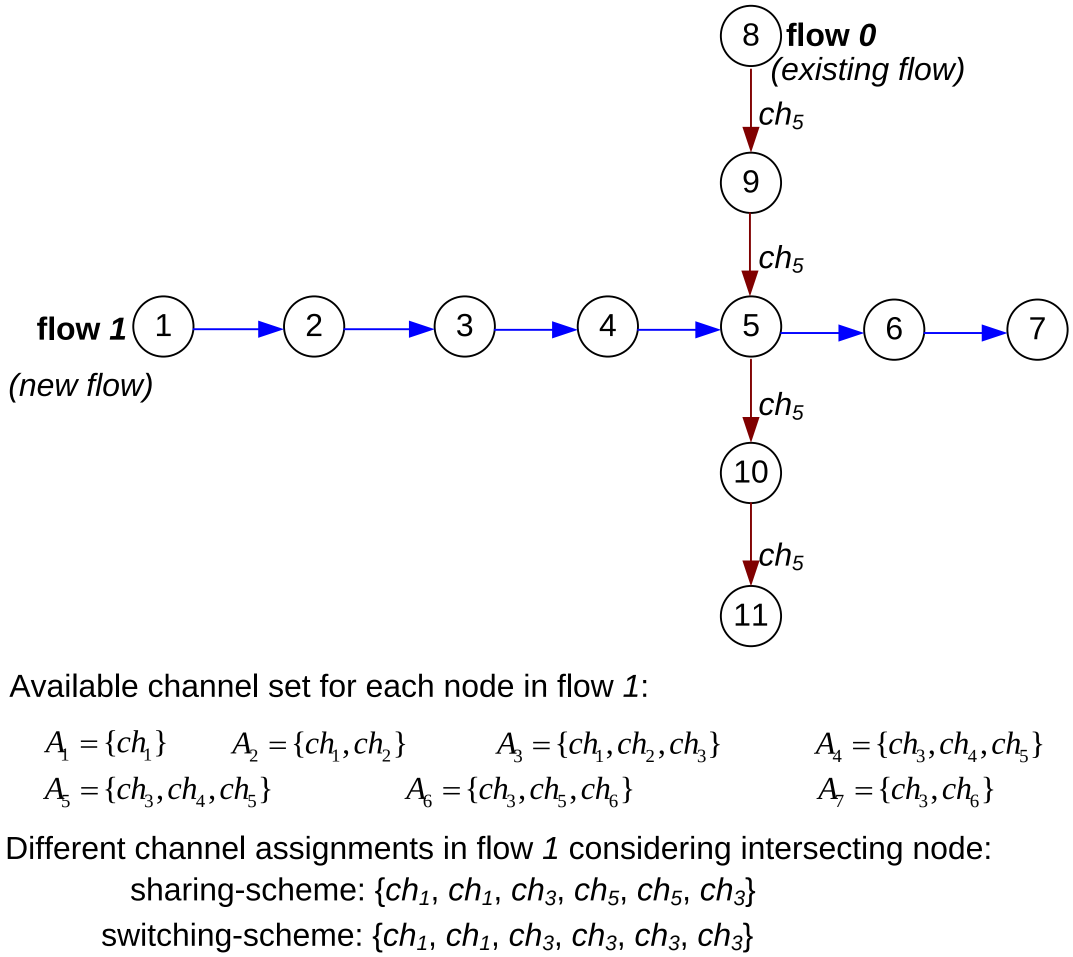


Figure 6.7: Different choices of operating channel at intersecting node when the new flow intersects with an existing flow

Figure 6.8) for the new flow 1, while the delays for existing flow 0 stay unchanged. Detailed investigations show that the sharing scheme has to switch to ch_5 at node 4, and then switch from ch_5 to ch_3 at node 6. These two extra switches are avoided by the MFS based switching scheme. In the switching scheme, node 5 switches to the operating channel of the other flow after it transmits a packet for one flow. With such scheduling the delay of channel switching at the intersecting node has a minimum impact on the end-to-end delay of both flows.

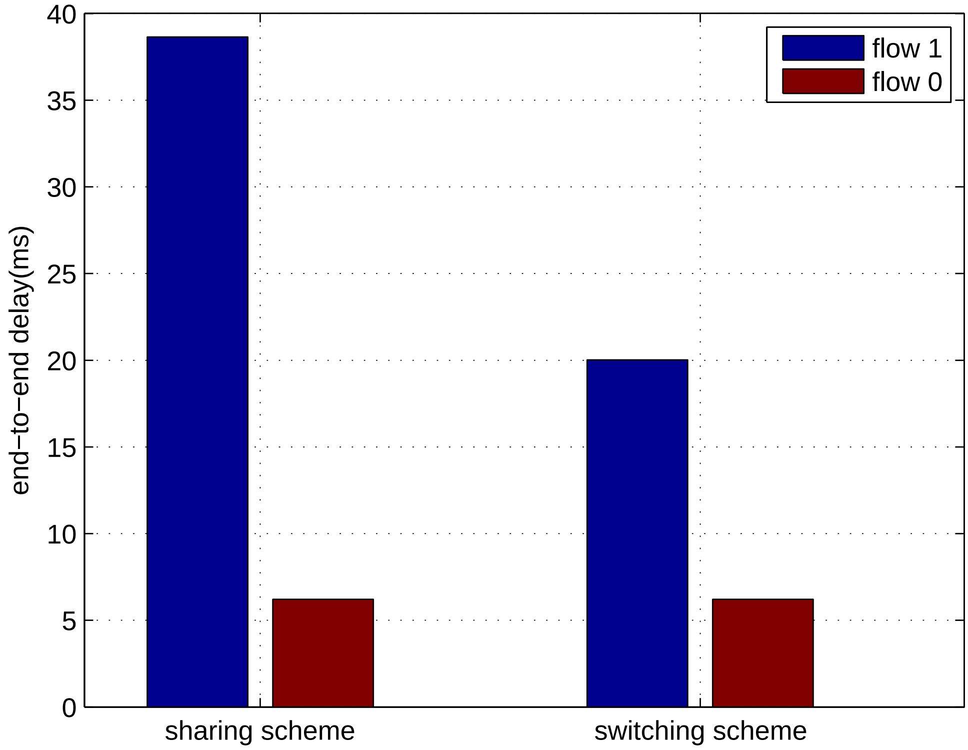


Figure 6.8: End-to-end delay of two flows under the two different channel selecting schemes

6.5 Summary

In this chapter, we proposed the MFS based channel assignment and routing algorithm that effectively reduced the number of channel switches in single transceiver, multi-hop cognitive radio networks. An efficient channel information dissemination mechanism was designed. Extensive ns-2 simulation results verified that our MFS based scheme delivered lower end-to-end delay and higher throughput than the link based approach. We also demonstrated the scalability with respect to the network size, the stability with respect to traffic load, as well as the reduced overhead of the proposed scheme. These indicate that the MFS based scheme is suitable for large scale multi-hop CRNs.

Chapter 7

Conclusions and Future Work

7.1 Conclusions

In this thesis, we have studied how to design effective routing algorithms in CRNs under DSA environments to improve end-to-end network performance. The research work concentrates on developing opportunistic routing and related multi-channel transmission schemes in highly dynamic spectrum CRNs, as well as an efficient channel assignment integrated with the routing scheme in single transceiver CRNs. Performance improvements of our proposed schemes are demonstrated via analytical and simulation results.

Firstly, in Chapter 3 we designed a spectrum aware opportunistic routing algorithm for multi-channel CRNs working in highly dynamic spectrum environments. The proposed opportunistic routing algorithm attempts to broadcast packets on all available channels of a CR link, fully exploiting the benefits of multiple channels. By considering practical primary user activity, channel fading and contention access among CR users, we integrate channel availability, successful transmission rate and successful channel access probability into our model to characterize the opportunistic nature of the CR links. The CR link consisting of multiple channels is further modeled as an M/Geo/1 queue. Our queuing analysis confirms the benefit of multi-channel transmission in terms of lower delay than single-channel transmission. A forwarder candidate list is setup based on the node delay metric derived from the link traffic model. Moreover, packet duplicates caused by broadcasting are avoided via network coding. The simulation results show that the proposed opportunistic routing algorithm achieves low end-to-end delay and supports high offered load. Moreover, the algorithm can compensate for lost performance due to lower channel availability by exploiting multiple channels.

Secondly, we investigated multi-channel batch transmission in single-hop and two-hop CRNs in a lossy environment in Chapter 4 and Chapter 5. In network coding based opportunistic routing protocols, batch transmission is a key component and its efficiency influences routing performance. In single-hop CRNs, a random linear coded batch transmission scheme is proposed in Chapter 4. In the proposed scheme, the source node sends random linear combinations of original packets in a batch, similar to the way it is done in random network coding. Therefore, the scheme can be easily extended to network coding in multi-hop networks. Compared to ARQ based schemes, the coded transmission scheme can reduce packet duplications during transmission and retransmission which makes packet transmission over multiple available channels more efficient. The associated batch transmission delay is derived and verified. The coded scheme experiences lower delay than the ARQ based schemes. As the coded scheme requires fewer acknowledgment messages, it is less dependent on feedback channels which may be scarce in CRNs. Integrated with an opportunistic routing strategy, the coded scheme is further extended for multi-channel batch transmission in a two-hop CRN in Chapter 5. The proposed scheme is compared with a traditional routing strategy based scheme. Our analyses, validated by simulations, show that the proposed scheme significantly outperforms the traditional one, reducing the batch delay by up to 60%.

Finally, we proposed a maximum flow-segment based channel assignment and routing algorithm in single transceiver CRNs in Chapter 6. The proposed scheme effectively reduces channel switches by maximizing the number of nodes that a selected channel can connect. The proposed algorithm experiences much lower end-to-end delay, with a reduction of 50% than link based schemes. Moreover, the proposed scheme achieves a higher maximum throughput. The channel information that needs to be disseminated is simplified via a K-tuple and channel information is processed in a distributed manner at intermediate nodes which reduces information exchange overhead and makes the scheme suitable for CRNs with less-functionalized nodes.

7.2 Future Work

In this thesis, we have investigated several critical issues in routing design for CRNs. There are several directions that our work can be expanded.

1. Routing design with limited number of radio interfaces.

In our multi-channel spectrum aware opportunistic routing design, CR users are assumed to be equipped with the same number of transceivers as the number of channels. However, in practice the number of radio interfaces on a CR user is normally smaller than the number of channels due to cost and implementation considerations. In such scenarios, the assignment of radio interfaces to channels has a significant impact on routing performance. Similar problems have been studied in traditional non-CR wireless network and CRNs with slow varying spectrum. However, this is still an open issue for opportunistic routing design in highly dynamic spectrum CRNs. One challenge in such an environment is how to coordinate common operating channels among neighbours with minimum overhead in the shortest time or even without frequent information exchange. Another problem is how to choose channels so that there are enough forwarder candidates that can receive packets for reliable forwarding. New solutions are expected for routing with a limited number of radio interfaces in multi-channel highly dynamic spectrum CRNs.

2. Flow scheduling based on forwarding contribution.

Given a node in a CRN, there can be multiple flows traversing the node. In traditional routing schemes with fixed routes the intersecting node must serve the different flows simultaneously and conduct a schedule between them. When working with opportunistic routing, the intersecting node is probably in the candidate lists of the different flows. However, there can be multiple forwarding candidates for the different flows in the surrounding area of the node which can forward packets instead of the specific intersecting node. When scheduling flows at the intersecting node, the contribution of this node to the different flows should be taken into account. For instance, one flow may have many forwarding candidates in the surrounding area while the other flow may only have the intersecting node as a forwarder. In such cases, the intersecting node can schedule more transmission opportunities for the second flow to improve transmission efficiency and overall network performance. Therefore, how to efficiently schedule multiple flows among multiple forwarding candidates according to their contributions should be further investigated.

3. Performance bound analysis of opportunistic routing in highly dynamic spectrum CRNs.

Theoretical analysis of the performance bound can provide a clear idea of how well opportunistic routing protocols can perform in highly dynamic spectrum CRNs. This bound sets a performance target for practical routing protocol design. Theoretical analysis can also provide ideas on how to improve the practical design. In this thesis, we have conducted theoretical analysis of the batch transmission delay in single-hop and two-hop CRNs, which provides insights into the performance understanding and design of opportunistic routing in CRNs. However, there is still a need to develop comprehensive analysis in general topology CRNs. There is some work reported in the literature on the analysis of opportunistic routing in traditional wireless networks. However, the dynamics of the available spectrum is not taken into account which has a significant impact on the routing performance in highly dynamic spectrum CRNs. To support more efficient opportunistic routing design comprehensive theoretical performance analysis is therefore highly desirable.

4. Combination of the current work and standardized MAC.

The work in this thesis was conducted under general contention based MACs. In fact, some work has been done for CR MAC standardization in the community, such as IEEE 802.22. IEEE 802.22 MAC is a specific MAC for wireless access with white space in TV bands. The general spectrum scenario is that the spectrum varies slowly. For the work in Chapter 6, it could be a good idea to recheck the algorithm combined with IEEE 802.22. Although the other work in this thesis focuses on dynamic spectrum environment, we still can try to do some work for the combination of standardized MAC. This would make our work more popular to the industry.

5. Transmission analysis in a star topology with CR base-station.

The work in this thesis considered some general topologies. A cell with one CR base-station and multiple CR mobile-stations is an another scenario. There are some works in the literature for non-CR star topology wireless networks and some of them employ network coding. It is a promising idea to extend our work in such a scenario, especially for the work in Chapter 4.

Appendix A

Abbreviations

ACK	Acknowledgment
ARQ	Automatic Repeat reQuest
AWGN	Additive White Gaussian Noise
CR	Cognitive Radio
CRN	Cognitive Radio Network
DSA	Dynamic Spectrum Access
EATT	Expected Anypath Transmission Time
EAX	Expected Any-path transmissions
EOT	Expected One-hop Throughput
ETX	Expected Transmission count
ExOR	Extremely Opportunistic Routing
FCC	Federal Communications Commission
FCFS	First Come First Served
FS	Flow-Segment
GBN-ARQ	Go-Back-N ARQ
MAC	Medium Access Control

MC-OLT	Multi-Channel Opportunistic Link Transmission metric
MFS	Maximum Flow-Segment
MORE	MAC-independent Opportunistic Routing and Encoding
MSAOR	Multi-channel Spectrum Aware Opportunistic Routing
NACK	Negative Acknowledgment
NAV	Network Allocation Vector
PHY	Physical
PU	Primary User
RF	Radio Frequency
RKRL	Radio Knowledge Representation Language
RREP	Route Reply
RREQ	Route Request
SAOR	Spectrum Aware Opportunistic Routing
SIFS	Short InterFrame Space
SR-ARQ	Selective-Repeat ARQ
SU	Secondary User
SW-ARQ	Stop-and-Wait ARQ
WLAN	Wireless Local Area Network

Bibliography

[1] J. Mitola III, "Cognitive radio for flexible mobile multimedia communications," in *IEEE International Workshop on Mobile Multimedia Communications (MoMuC)*, 1999, pp. 3–10.

[2] A. Goldsmith, *Wireless communications.* Cambridge university press, 2005.

[3] FCC, "Spectrum policy task force report," Tech. Rep., 2002.

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

[5] M. Cesana, E. Ekici, and Y. Bar-Ness, "Networking over multi-hop cognitive networks," *IEEE Network*, vol. 23, no. 4, pp. 4–5, 2009.

[6] S. Krishnamurthy, M. Thoppian, S. Venkatesan, and R. Prakash, "Control channel based MAC-layer configuration, routing and situation awareness for cognitive radio networks," in *IEEE Military Communications Conference (MILCOM)*, 2005, pp. 455–460.

[7] J. So and N. Vaidya, "A routing protocol for utilizing multiple channels in multi-hop wireless networks with a single transceiver," *Univ. Illinois at UrbanaChampaign (UIUC), Urbana, IL, Tech. Rep*, 2004.

[8] R. Vedantham, S. Kakumanu, S. Lakshmanan, and R. Sivakumar, “Component based channel assignment in single radio, multi-channel ad hoc networks,” in *Proceedings of the 12th annual international conference on Mobile computing and networking*, 2006, pp. 378–389.

[9] NTIA, “U.S. frequency allocations,” 2011. [Online]. Available: http://www.ntia.doc.gov/files/ntia/publications/spectrum_wall_chart_aug2011.pdf

[10] D. Cabric, I. D. O'Donnell, M. S. W. Chen, and R. W. Brodersen, “Spectrum sharing radios,” *IEEE Circuits and Systems Magazine*, vol. 6, no. 2, pp. 30–45, 2006.

[11] B. Wang and K. J. R. Liu, “Advances in cognitive radio networks: a survey,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 1, pp. 5–23, 2011.

[12] Y.-C. Liang, K.-C. Chen, G. Y. Li, and P. Mahonen, “Cognitive radio networking and communications: an overview,” *IEEE Transactions on Vehicular Technology*, vol. 60, no. 7, pp. 3386–3407, 2011.

[13] J. Mitola III and J. Maguire, G. Q., “Cognitive radio: making software radios more personal,” *IEEE Personal Communications*, vol. 6, no. 4, pp. 13–18, 1999.

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

[15] K. Shin, H. Kim, A. Min, and A. Kumar, “Cognitive radios for dynamic spectrum access: from concept to reality,” *IEEE Wireless Communications*, vol. 17, no. 6, pp. 64–74, 2010.

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

[17] B. Wild and K. Ramchandran, "Detecting primary receivers for cognitive radio applications," in *IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, 2005, pp. 124–130.

[18] R. Zhang and Y.-C. Liang, "Exploiting hidden power-feedback loops for cognitive radio," in *IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN 2008)*, 2008, pp. 1–5.

[19] G. Zhao, G. Y. Li, C. Yang, and J. Ma, "Proactive detection of spectrum holes in cognitive radio," in *IEEE International Conference on Communications (ICC)*, 2009, pp. 1–5.

[20] D. Cabric, S. M. Mishra, and R. W. Brodersen, "Implementation issues in spectrum sensing for cognitive radios," in *Conference Record of the Thirty-Eighth Asilomar Conference on Signals, Systems and Computers*, 2004, pp. 772–776.

[21] R. Tandra and A. Sahai, "Fundamental limits on detection in low SNR under noise uncertainty," in *International Conference on Wireless Networks, Communications and Mobile Computing*, 2005, pp. 464–469.

[22] D. Datla, R. Rajbanshi, A. M. Wyglinski, and G. J. Minden, "Parametric adaptive spectrum sensing framework for dynamic spectrum access networks," in *IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN 2007)*, 2007, pp. 482–485.

[23] J.-Y. Wu, C.-H. Wang, and T.-Y. Wang, “Performance analysis of energy detection based spectrum sensing with unknown primary signal arrival time,” *IEEE Transactions on Communications*, vol. 59, no. 7, pp. 1779–1784, 2011.

[24] K. Kyouwoong, I. A. Akbar, K. K. Bae, U. Jung-sun, C. M. Spooner, and J. H. Reed, “Cyclostationary approaches to signal detection and classification in cognitive radio,” in *IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, 2007, pp. 212–215.

[25] I. F. Akyildiz, W.-Y. Lee, and K. R. Chowdhury, “CRAHNs: Cognitive radio ad hoc networks,” *Ad Hoc Netw.*, vol. 7, no. 5, pp. 810–836, 2009.

[26] G. Ganesan and Y. Li, “Cooperative spectrum sensing in cognitive radio networks,” in *IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN 2005)*, 2005, pp. 137–143.

[27] C. Sun, W. Zhang, and K. B. Letaief, “Cooperative spectrum sensing for cognitive radios under bandwidth constraints,” in *IEEE Wireless Communications and Networking Conference (WCNC)*, 2007, pp. 1–5.

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

[29] Q. Zhao, L. Tong, and A. Swami, “Decentralized cognitive MAC for dynamic spectrum access,” in *IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, 2005, pp. 224–232.

[30] R. Etkin, A. Parekh, and D. Tse, “Spectrum sharing for unlicensed bands,” *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 3, pp. 517–528, 2007.

[31] L. Cao and H. Zheng, "Distributed spectrum allocation via local bargaining," in *IEEE SECON*, 2005, pp. 119–127.

[32] S. Sankaranarayanan, P. Papadimitratos, A. Mishra, and S. Hershey, "A bandwidth sharing approach to improve licensed spectrum utilization," in *IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, 2005, pp. 279–288.

[33] C. Peng, H. Zheng, and B. Zhao, "Utilization and fairness in spectrum assignment for opportunistic spectrum access," *Mobile Networks and Applications*, vol. 11, no. 4, pp. 555–576, 2006.

[34] Y. Wu, B. Wang, K. Liu, and T. Clancy, "Repeated open spectrum sharing game with cheat-proof strategies," *IEEE Transactions on Wireless Communications*, vol. 8, no. 4, pp. 1922–1933, 2009.

[35] C. Cormio and K. Chowdhury, "A survey on MAC protocols for cognitive radio networks," *Ad Hoc Networks*, vol. 7, no. 7, pp. 1315–1329, 2009.

[36] M. Cesana, F. Cuomo, and E. Ekici, "Routing in cognitive radio networks: Challenges and solutions," *Ad Hoc Networks*, vol. 9, no. 3, pp. 228–248, 2011.

[37] H. Khalif, N. Malouch, and S. Fdida, "Multihop cognitive radio networks: to route or not to route," *IEEE Network*, vol. 23, no. 4, pp. 20–25, 2009.

[38] S. Biswas and R. Morris, "ExOR: Opportunistic multi-hop routing for wireless networks," *ACM SIGCOMM Computer Communication Review*, vol. 35, no. 4, pp. 133–143, 2005.

[39] S. Chachulski, M. Jennings, S. Katti, and D. Katabi, “Trading structure for randomness in wireless opportunistic routing,” *SIGCOMM Comput. Commun. Rev.*, vol. 37, no. 4, pp. 169–180, 2007.

[40] H. Liu, B. Zhang, H. Mouftah, X. Shen, and J. Ma, “Opportunistic routing for wireless ad hoc and sensor networks: Present and future directions,” *IEEE Communications Magazine*, vol. 47, no. 12, pp. 103–109, 2009.

[41] C.-J. Hsu, H.-I. Liu, and W. K. G. Seah, “Opportunistic routing - a review and the challenges ahead,” *Computer Networks*, vol. 55, no. 15, pp. 3592–3603, 2011.

[42] K. Zeng, W. Lou, and H. Zhai, “Capacity of opportunistic routing in multi-rate and multi-hop wireless networks,” *IEEE Transactions on Wireless Communications*, vol. 7, no. 12, pp. 5118–5128, 2008.

[43] C.-P. Luk, W.-C. Lau, and O.-C. Yue, “An analysis of opportunistic routing in wireless mesh network,” in *IEEE International Conference on Communications (ICC)*, 2008, pp. 2877–2883.

[44] P. Jacquet, B. Mans, P. Muhlethaler, and G. Rodolakis, “Opportunistic routing in wireless ad hoc networks: upper bounds for the packet propagation speed,” *IEEE Journal on Selected Areas in Communications*, vol. 27, no. 7, pp. 1192–1202, 2009.

[45] K. Zeng, Z. Y. Yang, and W. J. Lou, “Opportunistic routing in multi-radio multi-channel multi-hop wireless networks,” *IEEE Transactions on Wireless Communications*, vol. 9, no. 11, pp. 3512–3521, 2010.

[46] R. Zheng and C. Z. Li, “How good is opportunistic routing? - a reality check under rayleigh fading channels,” *Ad Hoc and Sensor Wireless Networks*, vol. 10, no. 1, pp. 5–26, 2010.

[47] E. Rozner, J. Seshadri, Y. Mehta, and L. Qiu, "SOAR: Simple opportunistic adaptive routing protocol for wireless mesh networks," *IEEE Transactions on Mobile Computing*, vol. 8, no. 12, pp. 1622–1635, 2009.

[48] M. Pan, R. Huang, and Y. Fang, "Cost design for opportunistic multi-hop routing in cognitive radio networks," in *IEEE Military Communications Conference (MILCOM)*, 2008, pp. 1–7.

[49] S. Lin and K. Chen, "Spectrum aware opportunistic routing in cognitive radio networks," in *IEEE Global Telecommunications Conference (GLOBECOM)*, 2010, pp. 1–6.

[50] C. E. Perkins and E. M. Royer, "Ad-hoc on-demand distance vector routing," in *IEEE Workshop on Mobile Computing Systems and Applications (WMCSA)*, 1999, pp. 90–100.

[51] D. Johnson and D. Maltz, "Dynamic source routing in ad hoc wireless networks," *Mobile computing*, pp. 153–181, 1996.

[52] C. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," *ACM SIGCOMM Computer Communication Review*, vol. 24, no. 4, pp. 234–244, 1994.

[53] R. Draves, J. Padhye, and B. Zill, "Comparison of routing metrics for static multi-hop wireless networks," *ACM SIGCOMM Computer Communication Review*, vol. 34, no. 4, pp. 133–144, 2004.

[54] S. Biswas and R. Morris, "Opportunistic routing in multi-hop wireless networks," *ACM SIGCOMM Computer Communication Review*, vol. 34, no. 1, pp. 69–74, 2004.

[55] Y. Yan, B. Zhang, J. Zheng, and J. Ma, “CORE: a coding-aware opportunistic routing mechanism for wireless mesh networks,” *Wireless Commun.*, vol. 17, no. 3, pp. 96–103, 2010.

[56] W. Cedric, “Opportunistic routing in dynamic ad hoc networks: the oprah protocol,” in *IEEE International Conference on Mobile Adhoc and Sensor Systems (MASS)*, 2006, pp. 570–573.

[57] Y. Yuan, H. Yang, S. Wong, S. Lu, and W. Arbaugh, “ROMER: Resilient opportunistic mesh routing for wireless mesh networks,” in *IEEE Workshop on Wireless Mesh Networks (WiMesh)*, 2005.

[58] M. Zorzi and R. R. Rao, “Geographic random forwarding (GeRaF) for ad hoc and sensor networks: multihop performance,” *IEEE Transactions on Mobile Computing*, vol. 2, no. 4, pp. 337–348, 2003.

[59] K. Zeng, W. Lou, J. Yang, and D. Brown, “On throughput efficiency of geographic opportunistic routing in multihop wireless networks,” *Mobile Networks and Applications*, vol. 12, no. 5, pp. 347–357, 2007.

[60] Z. Zhong and S. Nelakuditi, “On the efficacy of opportunistic routing,” in *IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, 2007, pp. 441–450.

[61] R. Laufer, H. Dubois-Ferriere, and L. Kleinrock, “Multirate anypath routing in wireless mesh networks,” in *IEEE INFOCOM*, 2009, pp. 37–45.

[62] Y. Lin, B. Li, and B. Liang, “CodeOR: Opportunistic routing in wireless mesh networks with segmented network coding,” in *IEEE International Conference on Network Protocols (ICNP)*, 2008, pp. 13–22.

[63] Y. Lin, B. Liang, and B. Li, "SlideOR: Online opportunistic network coding in wireless mesh networks," in *IEEE INFOCOM*, 2010, pp. 1–5.

[64] D. Koutsonikolas, C.-C. Wang, and Y. C. Hu, "CCACK: Efficient network coding based opportunistic routing through cumulative coded acknowledgments," in *IEEE INFOCOM*, 2010, pp. 1–9.

[65] ——, "Efficient network-coding-based opportunistic routing through cumulative coded acknowledgments," *IEEE/ACM Transactions on Networking*, vol. 19, no. 5, pp. 1368–1381, 2011.

[66] B. Radunovic, C. Gkantsidis, P. Key, and P. Rodriguez, "Toward practical opportunistic routing with intra-session network coding for mesh networks," *IEEE/ACM Transactions on Networking*, vol. 18, no. 2, pp. 420–433, 2010.

[67] C. Xin, B. Xie, and C. Shen, "A novel layered graph model for topology formation and routing in dynamic spectrum access networks," in *IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, 2005, pp. 308–317.

[68] C. Xin, L. Ma, and C. Shen, "Path-centric channel assignment in cognitive radio wireless networks," in *International Conference on Cognitive Radio Oriented Wireless Networks and Communications (CrownCom)*, 2007, pp. 313–320.

[69] X. Zhou, L. Lin, J. Wang, and X. Zhang, "Cross-layer routing design in cognitive radio networks by colored multigraph model," *Wireless personal communications*, vol. 49, no. 1, pp. 123–131, 2009.

[70] Q. Wang and H. Zheng, "Route and spectrum selection in dynamic spectrum networks," in *IEEE Consumer Communications and Networking Conference (CNCC)*, 2006.

[71] G. Cheng, W. Liu, Y. Li, and W. Cheng, "Spectrum aware on-demand routing in cognitive radio networks," in *IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, 2007, pp. 571–574.

[72] H. Ma, L. Zheng, and X. Ma, "Spectrum aware routing for multi-hop cognitive radio networks with a single transceiver," in *International Conference on Cognitive Radio Oriented Wireless Networks and Communications (CrownCom)*, 2008, pp. 1–6.

[73] G. Cheng, W. Liu, Y. Li, and W. Cheng, "Joint on-demand routing and spectrum assignment in cognitive radio networks," in *IEEE International Conference on Communications (ICC)*, 2007, pp. 6499–6503.

[74] Z. Yang, G. Cheng, W. Liu, W. Yuan, and W. Cheng, "Local coordination based routing and spectrum assignment in multi-hop cognitive radio networks," *Mobile Networks and Applications*, vol. 13, no. 1, pp. 67–81, 2008.

[75] C. Zheng, R. Liu, X. Yang, I. Collings, Z. Zhou, and E. Dutkiewicz, "Maximum flow-segment based channel assignment and routing in cognitive radio networks," in *IEEE 73rd Vehicular Technology Conference (VTC Spring)*, 2011, pp. 1–6.

[76] G. Zhu, I. Akyildiz, and G. Kuo, "STOD-RP: a spectrum-tree based on-demand routing protocol for multi-hop cognitive radio networks," in *IEEE Global Telecommunications Conference (GLOBECOM)*, 2008, pp. 1–5.

[77] K. R. Chowdhury and I. F. Akyildiz, "CRP: A routing protocol for cognitive radio ad hoc networks," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 4, pp. 794–804, 2011.

[78] K. Chowdhury and M. Felice, "Search: a routing protocol for mobile cognitive radio ad-hoc networks," *Computer Communications*, vol. 32, no. 18, pp. 1983–1997, 2009.

[79] A. Sampath, L. Yang, L. Cao, H. Zheng, and B. Zhao, "High throughput spectrum-aware routing for cognitive radio networks," *IEEE Crowncom*, 2008.

[80] H. Khalife, S. Ahuja, N. Malouch, and M. Krunz, "Probabilistic path selection in opportunistic cognitive radio networks," in *IEEE Global Telecommunications Conference (GLOBECOM)*, 2008, pp. 1–5.

[81] Y. Wen and W. Liao, "On QoS routing in wireless ad-hoc cognitive radio networks," in *IEEE 71st Vehicular Technology Conference (VTC 2010-Spring)*, 2010, pp. 1–5.

[82] X. Huang, G. Wang, F. Hu, and S. Kumar, "Stability-capacity-adaptive routing for high-mobility multihop cognitive radio networks," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 6, pp. 2714–2729, 2011.

[83] K. Bian and J. Park, "Segment-based channel assignment in cognitive radio ad hoc networks," in *International Conference on Cognitive Radio Oriented Wireless Networks and Communications (CrownCom)*, 2007, pp. 327–335.

[84] I. Pefkianakis, S. Wong, and S. Lu, "SAMER: spectrum aware mesh routing in cognitive radio networks," in *IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, 2008, pp. 1–5.

[85] C. Zheng, E. Dutkiewicz, R. Liu, R. Vesilo, G. Fang, and Z. Zhou, "Opportunistic routing in multi-channel cognitive radio networks," in *International Symposium on Communications and Information Technologies (ISCIT)*, 2011, pp. 532–537.

[86] A. Raniwala and C. Tzi-cker, "Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network," in *IEEE INFOCOM*, 2005, pp. 2223–2234.

[87] A. H. Mohsenian-Rad and V. W. S. Wong, "Joint logical topology design, interface assignment, channel allocation, and routing for multi-channel wireless mesh networks," *IEEE Transactions on Wireless Communications*, vol. 6, no. 12, pp. 4432–4440, 2007.

[88] S. Wang, J. Zhang, and L. Tong, "Delay analysis for cognitive radio networks with random access: A fluid queue view," in *INFOCOM*, 2010, pp. 1–9.

[89] V. Pla, S. De Vuyst, K. De Turck, E. Bernal-Mor, J. Martinez-Bauset, and S. Wittevrongel, "Saturation throughput in a heterogeneous multi-channel cognitive radio network," in *IEEE International Conference on Communications (ICC)*, 2011, pp. 1–5.

[90] D. Bertsekas, R. Gallager, and P. Humblet, *Data networks.* Prentice-Hall International, 1992, vol. 2.

[91] J. F. Chang and T. H. Yang, "Multichannel ARQ protocols," *IEEE Transactions on Communications*, vol. 41, no. 4, pp. 592–598, 1993.

[92] Z. Ding and M. Rice, "ARQ error control for parallel multichannel communications," *IEEE Transactions on Wireless Communications*, vol. 5, no. 11, pp. 3039–3044, 2006.

[93] J. Li and Y. Zhao, "Packet delay analysis for multichannel communication systems with MSW-ARQ," *Performance Evaluation*, vol. 66, no. 7, pp. 380–394, 2009.

[94] J. Li and Y. Q. Zhao, "Resequencing analysis of stop-and-wait ARQ for parallel multichannel communications," *IEEE/ACM Transactions on Networking*, vol. 17, no. 3, pp. 817–830, 2009.

[95] Y. Chung and Z. Tsai, "Performance analysis of two multichannel fast retransmission schemes for delay-sensitive flows," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 7, pp. 3468–3479, 2010.

[96] H. Seferoglu, A. Markopoulou, and K. Ramakrishnan, "I^{2}NC: Intra-and inter-session network coding for unicast flows in wireless networks," in *IEEE INFOCOM*, 2011, pp. 1035–1043.

[97] T. Ho, M. Mdard, R. Koetter, D. Karger, M. Effros, J. Shi, and B. Leong, "A random linear network coding approach to multicast," *IEEE Transactions on Information Theory*, vol. 52, no. 10, pp. 4413–4430, 2006.

[98] P. Chou, Y. Wu, and K. Jain, "Practical network coding," in *Proceedings of the Annual Allerton Conference on Communication Control and Computing*, 2003, pp. 40–49.

[99] D. Lun, M. Mdard, R. Koetter, and M. Effros, "On coding for reliable communication over packet networks," *Physical Communication*, vol. 1, no. 1, pp. 3–20, 2008.

[100] T. Ho and D. Lun, *Network coding: an introduction.* Cambridge University Press, 2008.

[101] H. Kushwaha, Y. Xing, R. Chandramouli, and H. Heffes, "Reliable multimedia transmission over cognitive radio networks using fountain codes," *Proceedings of the IEEE*, vol. 96, no. 1, pp. 155–165, 2008.

[102] J. K. Sundararajan, D. Shah, and M. Mdard, "ARQ for network coding," in *IEEE International Symposium on Information Theory (ISIT)*, 2008, pp. 1651–1655.

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

[104] C. Fragouli, J. Le Boudec, and J. Widmer, "Network coding: an instant primer," *ACM SIGCOMM Computer Communication Review*, vol. 36, no. 1, pp. 63–68, 2006.

[105] "The network simulator - ns-2." [Online]. Available: http://www.isi.edu/nsnam/ns/

[106] R. Calvo and J. Campo, "Adding multiple interface support in ns-2," 2007. [Online]. Available: http://telecom.inescporto.pt/~rcampos/ucMultiIfacesSupport.pdf

[107] B. Wang, "NS2 notebook: multi-channel multi-interface simulation in NS2 (2.29)." [Online]. Available: http://www.cse.msu.edu/~wangbo1/ns2/nshowto8.html

[108] "Cognitive radio cognitive network simulator." [Online]. Available: http://stuweb.ee.mtu.edu/~ljialian/

[109] P. Li, N. Scalabrino, Y. Fang, E. Gregori, and I. Chlamtac, "How to effectively use multiple channels in wireless mesh networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 20, no. 11, pp. 1641–1652, 2009.