THE HONG KONG POLYTECHNIC UNIVERSITY
DEPARTMENT OF COMPUTING

CROSS-LAYER OPTIMIZATION OF WIRELESS NETWORKS

By
Wei Feng

A thesis submitted in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy
May 2011

CERTIFICATE OF ORIGINALITY

Date: **May 2011**

Author: **Wei Feng**

Title: **Cross-layer Optimization of Wireless Networks**

Department: **Department of Computing**

Degree: **Ph.D.** Convocation: **August 10** Year: **2011**

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgement has been made in the text.

Signature of Author Wei Feng

Abstract

Existing wireless network is designed based on the layered network structure. This strict layered network structure limits the flexibility of protocol design. In this dissertation, we investigate the methodology of cross-layer optimization and utilize it to optimize various wireless networks. Specifically speaking, we make three original contributions in this dissertation.

First, we design and implement the first cooperative QoS routing protocol in cooperative multi-hop wireless network. Existing works about cooperative QoS routing did not consider the interference effect among links, and only evaluate their algorithms through simulations. This paper targets at designing and implementing an interference-aware Cooperative QoS routing protocol (CQ-routing) in the real testbed. We formulate the problem of finding cooperative routing path with maximum available bandwidth as an optimization problem, called Coop-routing problem. We prove that the Coop-routing problem is strong NP-hard. We propose both centralized and distributed approximation algorithms to solve the Coop-routing problem. We design and implement a CQ-routing protocol in wireless mesh network testbed and evaluate its performance through both experiments and simulations. The results show that CQ-routing protocol can significantly improve the network performance in terms of available bandwidth and number of admitter flows.

Second, we propose the first multi-link spectrum handoff scheduling algorithms for multi-hop cognitive network. Existing work only considered the problem of minimizing spectrum handoff delay of a single link in single-hop cognitive networks, referred to as the SH-SLSH problem. We study a more challenging problem in which multiple links perform spectrum handoff in multi-radio multi-hop cognitive networks (referred to as the SH-MLMH problem). The SH-MLMH problem targets at minimizing the Total Handoff Completion Time (THCT) while maintaining the network connectivity. The THCT is defined as the time for all the links to finish spectrum handoff. We prove that SH-MLMH problem is NP-hard. We propose both centralized and distributed algorithms to solve it. The simulation results show that our proposed solution can significantly improve the network throughput and reduce the THCT.

Third, we study the non-cooperative channel and bandwidth allocation problem in multi-radio multi-channel wireless network. Existing works ignored two important issues, impact of traffic load to channel's transmission quality, and difference of bandwidth demands for different node pairs. To address these two issues, we extend the problem of non-cooperative multi-radio channel allocation to Non-cooperative Joint Channel and Bandwidth Allocation problem (NJCBA), in which node pairs need to consider not only allocating radios to channels, but also allocating bandwidth to selected channels to maximize its own benefit. We prove that there exist pure Nash Equilibriums (NEs) for the NJCBA game. We also analyze the efficiency of the NEs for NJCBA game, and prove that these NEs can achieve a constant Price Of Anarchy (POA). We design two distributed algorithms, to enable node pairs to converge to a pure NE. The simulation results show that these two algorithms can improve the system throughput by 2 or 3 times compared with a greedy allocation algorithm.

Publications

Journal Paper

1. **Wei Feng**, Jiannong Cao, Liang Yang, Qin Xin, *"A Cross-layer Cooperative QoS Routing for Multi-hop Wireless Networks"*, submitted to the IEEE Transactions on Parallel and Distributed Systems (TPDS) on July 15th, 2011.
2. **Wei Feng**, Jiannong Cao, Chisheng Zhang, Qin Xin, *"Scheduling of Multi-link Spectrum Handoff in Multi-radio Multi-hop Cognitive Networks"*, submitted to Journal of Parallel and Distributed Computing (JPDC) on July 15th, 2011.
3. Bo Wang, **Wei Feng**, *"A Concurrent Multi-path Transfer Protocol used in Ad Hoc Networks"*, in proceedings of IET Journal of communications, 2010, Vol: 4, PP: 884 - 893.

Conference Paper

1. **Wei Feng**, Jiannong Cao, Chisheng Zhang, Chuda Liu, *"Joint Optimization of Spectrum Handoff Scheduling and Routing in Multi-hop Multi-radio Cognitive Networks"*, in proceedings of the 29th IEEE International Conference on Distributed Computing Systems, 2009, ICDCS '09, Montreal, Canada.

2. **Wei Feng**Jiannong CaoLiang Yang, *"Non-cooperative Quality-Aware Channel and Bandwidth Allocations in Multi-radio Multi-channel Wireless Networks"*, in proceedings of IEEE WCNC 2011, Cancun, Mexico.

3. Qin Xin, Xin Wang, Jiannong Cao, **Wei Feng**, "Joint Admission Control, Channel Assignment and QoS Routing for Coverage Optimization in Multi-hop Cognitive Radio Cellular Networks", in proceedings of the 8th IEEE International Conference on Mobile Ad-hoc and Sensor Systems (IEEE MASS 2011).

4. Jiannong Cao, Kun Xie, Weigang Wu, Chuda Liu, Gang Yao, **Wei Feng**, Yang Zou,"HAWK: Real-world Implementation of High-performance Heterogeneous Wireless Network for Internet Access", in proceedings of the 1st ICDCS International Workshop on Next Generation Network Architectures (NGNA'09), Montreal, Canada.

5. **Wei Feng**, Wenhong Liu, Huachun Zhou, Hongwei Huo, "The technology of route optimization for NEMO based on DHCPPD", in proceedings of 2006 IET International Conference on Wireless, Mobile and Multimedia Networks, WMN'06, hangzhou, China.

Patents

1. **Wei Feng**, Jiannong Cao, *"Multi-Subcarrier Scheduling for Throughput Maximization in Multi-Hop OFDMA Wireless Mesh Networks"*, in application.

2. **Wei Feng**, Jiannong Cao, *"A new billing method for cognitive wireless networks"*, in application.

Acknowledgements

I would like to thank many people who have helped me, and taken care of me. In particular, I am really grateful to my supervisor, Prof Jiannong Cao, for his guidance about my study and care about my life. He is not only a good researcher with broad knowledge but also a nice friend and kind person. He has sharp intuition, and grand insight about research, and taught me a lot about how to be a good researcher. He is such a nice person that anybody would like to make friends with him. I will always appreciate his kindness and warmheart.

I would also like to thank my girl friend, Yuling Zhang for her patience, waiting and love. We have been separated for more than three years until now. However, she has never complained me about this separation. Instead, she always encourages me and takes care of my parents. I believe that she is the best woman for me and I will spend the rest of my life with her.

I would also like to show my gratitude to my family, including my parents, my elder brother, and my uncles. They have brought me up with their love, and taught me a lot about how to be a good person. They have also sacrificed a lot for me. Without them, I can achieve nothing in this world. I will always remember what they did to me.

I also want to thank my roommates, Mr. Liang Yang and his family. I spent a

lot of happy time with him. His wife, Ning Fei, is also a nice person with warmheart. I wish them happy for ever. I also want to thank Dr. Qin Xin and Mr Chisheng Zhang because they help me a lot with my research works. I also want to show my gratitude to my friends, Jie Zhou, Miao Xiong, and Jing Jing Li. They are all very cute girls. I wish them beautiful and happy for ever. I also want to thank my friends, Hongjian Li, Chao Yang, Lei Yang, Qinjun Xiao, Chuda Liu, and Yang Zou. They are all good people. I wish them happy for ever. I also want to show my gratitude to my colleagues, Jun Zhang, Nianbo Liu, Xianlong Jiao, Yao Yu, Xin Xiao. Working with them is a valuable experience for me.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

In this chapter, we discuss the idea of cross-layer optimization, including its concept, advantages, and main methodologies. In section 1.1, we introduce the development of wireless networks, requirements for new wireless networks, and motivations for cross-layer optimization methodology. In section 1.2, we introduce the concept of cross-layer optimizations. In section 1.3, we summarize the principles for cross-layer optimizations. In section 1.4, we propose a unified cross-layer optimization framework for the dissertation. In section 1.5, we summarize the main contributions of this dissertation. In section 1.6, we point out the impacts of our research. In section 1.7, we describe the organization of this dissertation.

1.1 Next-generation Wireless Networks

Wireless communication technology has significantly changed people's life. Using wireless device, people can communicate with each other in any time and any place. However, the growing demand for wireless communication have following new requirements.

- Better QoS support: This can be attributed to high demand for wireless multimedia services such as data, voice, video, and the development of new wireless standards. The multimedia transmission requires us to provide communication services with guaranteed performance, such as bandwidth, delay. However, it is very hard to provide QoS-guaranteed transmission in wireless network because of the interference effect and variable wireless link condition. This requires us to reconsider the design of protocol stack in wireless network.

- High spectrum efficiency: The number of wireless devices have increased exponentially because of wide applications of wireless communication. However, the spectrum resource is very limited, so the development of wireless network makes spectrum become very scare resource. What is more worse, spectrum resource is allocated statically according to current regulations. The static spectrum allocation rules lead to low spectrum efficiency. This situation requires to design new technology to improve spectrum efficiency.

- Small handoff delay: Handoff refers to the operation of switching operating frequency. It can be caused by location mobility or spectrum mobility. Handoff can cause communication break, in terms of large handoff delay. This will make inconvenience to mobile users.

Moreover, another feature of next-wireless network is appearance of multiple new advanced communication technology . Specifically speaking, in this dissertation, our research is related to three new physical communication technologies, including cooperative communication technology, cognitive radio technology, and multi-radio multichannel technology. In cooperative communication technology, multiple helper nodes can cooperate with each other, and construct a virtual antenna array to help poor links transmit packets. In this way, the cooperative communication technology can improve poor links' transmission capacity. Cognitive radio is a new physical technology to solve the frequency scarcity problem through dynamic spectrum access. In cognitive networks, there are two kinds of users, primary users and secondary users. Primary users have the license to access the spectrum, while secondary users do not have the license to access the spectrum. By using cognitive radio technology, secondary users can use the licensed spectrum of primary users under the constraint of not interfering primary user s' communication. In this way, cognitive radio technology can improve the spectrum efficiency. Multi-radio multi-channel is another new physical technology, in which each wireless node is equipped with multiple interfaces working on different orthogonal channels. In this way, multiple interfaces can work simultaneously by tuning their interfaces working on different orthogonal channels, so the system throughput can be improved.

The new physical technologies bring both opportunities and challenges. In one side, these technologies are physical technologies, so we need to redesign upper layer protocols to fully explore their potentials. This requires us to violate the layered network structure, and adopt cross-layer optimization methodologies to optimize the performance of wireless networks. However, existing network structure is designed by layers, such as OSI seven layer network structure. The layered network structure is too rigid, which limits the performance of wireless networks. This requires us to propose new optimization methods, which could violate the layered network structure and optimize the wireless network in a cross-layer way. In the other side, these new physical technologies also cause some new problems, such as spectrum handoff, which will be discussed later. These new problems require us to design new mechanisms to handle them. These mechanism is not only related to protocols in single layer, but protocols in multiple layers. So we also need to handle them using cross-layer way.

In this dissertation, we target at utilizing cross-layer optimization methodologies to optimize the performance of next-generation wireless networks. In the next-generation wireless network, we will consider multiple new physical technologies, including cooperative communication technology, cognitive radio technology, multi-radio multi-channel technology. We hope that this research will promote the development of next-generation wireless networks.

1.2 Cross-layer Optimization Concept

In this subsection, we investigate the cross-layer optimization methodology. We first introduce existing layered network structure, and analyze its disadvantages. To overcome these disadvantages, we introduce the concept of cross-layer optimization. We provide a formal definition for cross-layer optimization. Then we explain the motivation for cross-layer optimization. Finally, we also introduce three main cross-layer optimization methodologies. Traditionally, network protocols are divided into several independent layers. Each layer is designed separately, and the interaction between layers is performed through a well-defined interface. The most well-known layered structure is the OSI reference model proposed by the International Organization for Standardization (ISO) in 1991, [Bertsekas and Gallager, 1992], as shown in Fig.1.1.

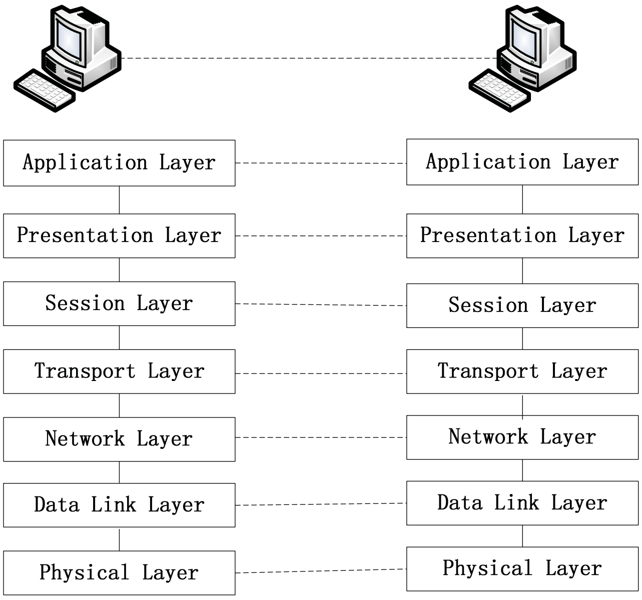


Fig. 1.1: The OSI reference model.

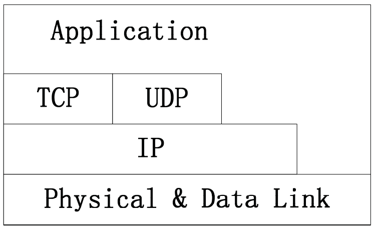


Fig. 1.2: The TCP/IP protocol stack.

There are seven layers in the OSI reference model, and each layer is responsible for one separated functionality. Based on the OSI reference model, DARPA (an agency of the United States Department of Defense) proposed the TCP/IP network protocol stack in the 1970s, [Bertsekas and Gallager, 1992]. The TCP/IP protocol stack is shown in Fig.1.2. The TCP/IP protocol stack combines three top layers in the OSI model the Application Layer, the Presentation Layer and the Session Layer, which are not distinguished separately in the TCP/IP model where it is just the Application

Layer. The main advantage of the layered network structure is architectural flexibility. In the layered network structure, different functions are separated among different layers. Each function realized inside a layer has to be performed independently of all other layers. In this way, we can easily replace the protocol in one layer without modifying protocols in other layers.

However, in actual networks, functions realized at different layers in the protocol stack are interdependent with each other, and they need to interact with each other in a complicated method. Although the OSI reference model leads to an easy design of the basic network functionalities, it imposes great limitations on the performance optimization. The main disadvantage of layered network structure is its rigid network structure. Each layer only cares about its functionality and layers directly above or below it. This feature ignores the interaction between different layers. It will lead to sub-optimal network performance. For example, it is hard for us to provide QoS support and mobility management in the layered network structure since the QoS support and mobility management require interaction between different layers as shown in Fi.g 1.3. From above analysis, we find that the layered network structure limits the optimization of the network performance, so we need to break the layered structure and optimize the network performance in a cross-layer way.

We first provide a definition of cross-layer optimization methodology. After we review literatures in this area, we find that there are several interpretations of cross-layer optimization. This is probably because the cross-layer optimization effort has been made by researchers from different backgrounds and different layers of the stack. We adopt the definition in [Bertsekas and Gallager, 1992]. Although this definition is simple and arguably obvious, we think that this definition serves to unify the different interpretations of cross-layer optimization in the literature. The formal definition of cross-layer optimization is described as follows.

Definition 1. (Cross-layer Optimization) Protocol design by the violation of a reference layered communication architecture is cross-layer design with respect to the particular layered architecture.

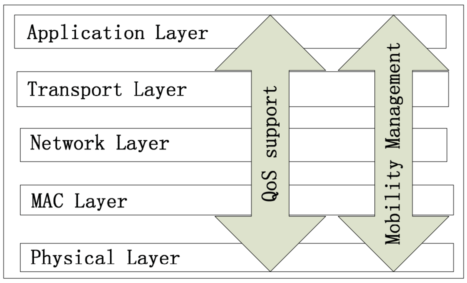


Fig. 1.3: The cross-layer protocol stack.

From above definition, we can find that cross-layer optimization refers to protocol design by violating the reference architecture, for example, by allowing direct communication between protocols at nonadjacent layers or sharing variables between layers.

There are three main reasons for us to optimize the performance of wireless networks in a cross-layer way. First, the layered network structure can not handle several problems caused by wireless links. For example, in the wireless network, the TCP sender mistakes a packet error on a wireless link as an indicator of network congestion. Moreover, in the wireless network, we need to consider mobility management issue. The mobility management requires us to optimize the network performance in a cross-layer method. However, the layered network structure can not handle this well. Second, the layered network structure did not consider opportunistic transmission of wireless medium. For example, the time-varying link quality allows opportunistic usage of the channel , in which the transmission parameters can be dynamically adjusted according to the variations in the channel quality. Third, the layered network structure did not consider broadcast nature of wireless medium. The nodes can also make use of the broadcast nature of the channel and cooperate with one another to minimize the interference. Therefore, we need to optimize the wireless network in a cross-layer way.

1.3 Principles of Cross-layer Optimization

In this section, we summarize the principles for cross-layer optimization, including when cross-layer optimization should be invoked and how the cross-layer optimization should be performed.

We summarize when cross-layer optimization should be invoked in two cases. The first case refers to the protocol design which needs to share information in a cross-layer way. More specifically, we need to design protocols which requires direct communication between protocols at nonadjacent layers or sharing variables between layers. For example, we may design routing protocol in the network layer, which needs information from the physical layer. The second case refers to the protocol design which jointly adjusts parameters in different layers. In this case, protocols in different layers may collaborate to achieve one objective. For example, we may design a protocol, which jointly consider routing and channel allocation to maximize the throughput. This requires us to jointly adjust parameters in network and physical layers.

We summarize the methods for cross-layer optimization as follows.

- Creation of new interfaces: We create new interfaces to share information between different layers in Fig. 1.4. According to the direction of information flow, we can further classify the method of creating new interfaces into three categories: upward, downward, back and forth information flow. In the example shown in Fig. 1.4, the transport layer is sharing information with physical layer by creating new interfaces.

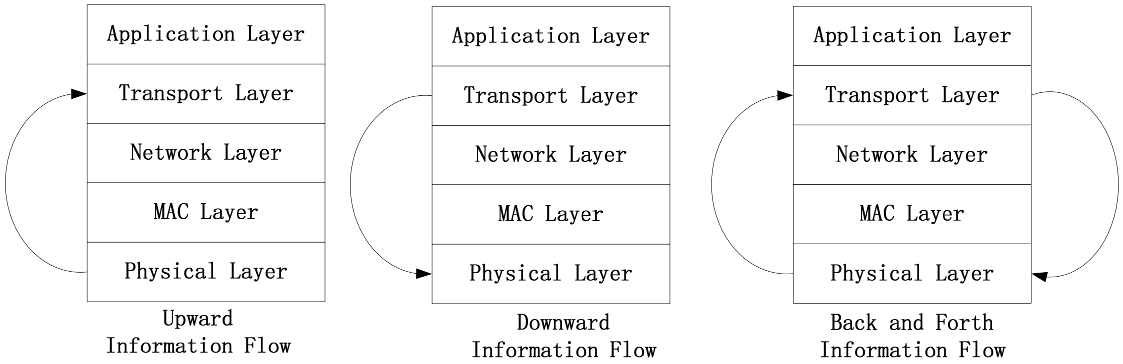


Fig. 1.4: Cross-layer optimization by creating new interface.

- Vertical calibration: The vertical calibration implies that we adjust parameters that span across layers because the final performance of application depends on parameters of multiple layers. There are two kinds of vertical calibrations, including static calibration and dynamic calibration. For static calibration, the parameters across layers can be set during design time. In this method, we do not necessarily need to design new interfaces between layers. For dynamic calibration, it requires a flexible protocol stack that responds to variations in the channel, traffic, and overall network conditions. For example, [Tong et al., 2004] considers the design of a MAC layer for the uplink of a wireless LAN when the PHY is capable of providing multi-packet reception capability. Because multi-packet reception capability enables the PHY to receive more than one packet at the same time, this changes the role of the MAC layer. Therefore, it needs to be redesigned. We need to jointly design protocols in physical and MAC layer. Parameters in both physical and MAC layers need to be tuned together.

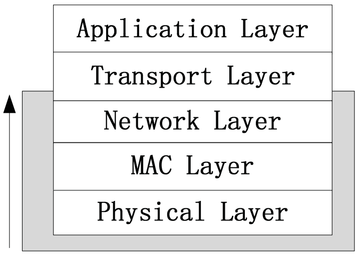


Fig. 1.5: Cross-layer optimization by vertical calibration.

- Completely new abstractions: Some researchers proposed revolutionary ideas, which completely reconstruct the network structure. [Tong et al., 2004] proposed a new way to organize the protocols: in heaps, not in stacks as done by layering. However, this method is not compatible with existing network structure, so it is not practical in the real wireless network.

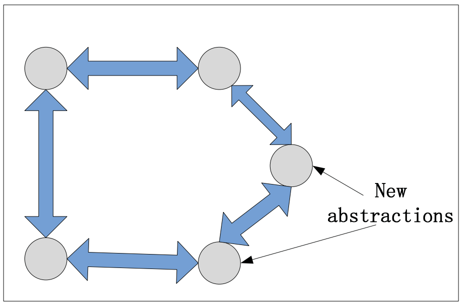


Fig. 1.6: Complectly new abstractions.

1.4 A Unified Cross-layer Optimization Framework

In this section, we propose a unified cross-layer optimization framework for optimizing the next-generation wireless networks. The framework of cross-layer optimization in this dissertation is described in Fig. 1.7. From the framework, we can find that we study three different physical technologies, including cooperative communication technology, the cognitive radio technology and multi-radio multi-channel technology. We will study three issues for these three physical technologies.

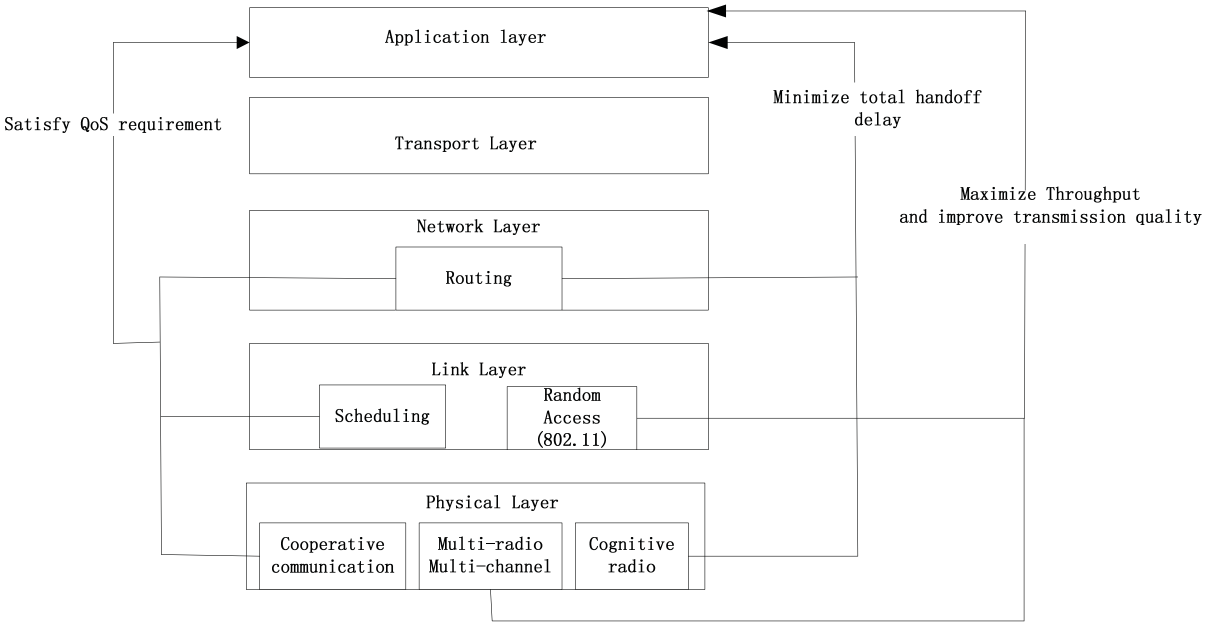


Fig. 1.7: Cross-layer Optimization Framework.

First, we study the cooperative QoS routing in multi-hop wireless network. In this issue, we target at utilizing cooperative communication technology to provide QoS support for multi-hop wireless network. Specifically speaking, we target at designing a cross-layer cooperative QoS routing protocol to satisfy users' bandwidth requirement through cooperative communication technology. From Fig. 1.7, we find that three issues in three different layers are considered, including cooperative communication technology in physical layer, scheduling in MAC layer and routing in network layer. We adopt the first cross-layer optimization method described in above subsection. We define back and forth interfaces between network, MAC and physical layer. The physical layer will send the link information, like SINR, to the routing protocol through this interface. Based on this collected information, the routing protocol will select both routing path and helper nodes for cooperative transmission. The routing protocol will send information about selected helpers to the physical layer.

Second, we study multi-link spectrum handoff problem in multi-hop cognitive networks. From 1.7, we consider two issues in two layers, the routing issue in the network layer and the spectrum handoff in the physical layer. We define an upward interface between these two layers. We formulate a new optimization problem, denoted by SH-MLMH problem. The SH-MLMH problem targets at deciding the order of spectrum handoff such that the Total Handoff Completion Time (THCT) is minimized while maintaining the network connectivity. We will design both centralized and distributed algorithms, to solve the SH-MLMH problem. The scheduling algorithms for the SH-MLMH problem is also responsible for coordination routing protocol to find alternative routing paths. The scheduling algorithms will send the scheduling result to the routing protocol to trigger routing update through the defined interface.

Third, we study channel assignment and bandwidth allocation in multi-radio multi-channel wireless network. In this work, we adopt the second cross-layer design method of vertical calibration. Specifically speaking, we calibrate channel assignment in the physical layer with bandwidth allocation in the MAC layer. Each node adjusts parameters of channel in physical layer and bandwidth in MAC layer together such that it can get good transmission quality. We model this problem as a non-cooperative static game, and analyze the final stable state.

In these three works, we explore two different kinds of cross-layer optimization methods, including creating interface and vertical collaboration. Because the third method is not practical, we do not discuss it in the study. In the first work, we use the method of creating interfaces between different layers. More specifically, we define back and forth interfaces between network, MAC and physical layers. In the upward direction, the protocols in MAC and physical layers collect link information and send them to the routing protocol in the network layer. In the downward direction,the routing protocol sends the information about selected helper nodes to protocols in the MAC and physical layers. In the second work, we also use the method of creating interfaces between different layers. More specifically, we define upward interfaces between network, MAC and physical layers. The coordination scheme for multi-link spectrum handoff in the physical layer sends information about coordination results to routing protocol in the network layer. The routing protocol performs rerouting according to the results of multi-link spectrum handoff. The difference between these two interfaces is that the first interface is bi-directional while the second one is u-nidirectional. In the third work, we use the method of vertical calibration. More specifically, we jointly adjust two parameters in physical and MAC layers. In the physical layer, we adjust the parameter of channel allocation for multiple radios. In the MAC layer, we adjust the parameter of bandwidth allocation for selected channels. The results of this study show that the cross-layer optimization method have significant advantages over the layered method.

1.5 Contribution of the Dissertation

In this dissertation, we target at optimizing the performance of wireless network using cross-layer optimization methodology. We study three issues in three different wireless networks, and summarize main contributions in this dissertation as follows.

For the first issue, we study the cooperative QoS routing in multi-hop wireless network. The main contribution of this issue is that we design the first cooperative QoS routing protocol (CQ-routing protocol), and implement it in the real test bed. Specifically speaking, we make following main contributions. CQ-routing protocol works with available bandwidth as routing metric, and it targets at satisfying users' bandwidth requirement by finding the routing path with maximum available bandwidth. We formulate the problem of finding routing path with maximum available bandwidth as an optimization problem, called Coop-routing problem. Given a new connection request in a cooperative multi-hop wireless network, Coop-routing problem targets at finding a routing path P and corresponding helper nodes and scheduling scheme for links in P such that P can achieve the maximum available bandwidth and existing flows are free from interference. We prove that the Coop-routing problem is a strong NP-hard problem. We propose both centralized and distributed polynomial approximation algorithms to solve the Coop-routing problem. We prove that the centralized algorithm can achieve a constant approximation ratio. Based on the distributed algorithm, we design the Cooperative QoS routing protocol (CQ-routing) to provide QoS support. We implement CQ-routing protocol on our wireless mesh network testbed and evaluate its performance via experiments. The results show that CQ-routing protocol can significantly improve the network performance in terms of available bandwidth and admission ratio.

For the second issue, we study multi-link spectrum handoff in multi-hop cognitive networks. Existing work only considered the problem of minimizing spectrum handoff delay of a single link in single-hop cognitive networks, referred to as the SH-SLSH problem. The objective of minimizing single link's spectrum handoff delay is not applicable for a more challenging problem (referred to as the SH-MLMH problem) in which multiple links perform spectrum handoff in multi-radio multi-hop cognitive networks. Instead of minimizing single link's spectrum handoff delay, the SH-MLMH problem first targets at maintaining the network connectivity so as to keep the communication uninterrupted during spectrum handoff. This results in eliminating the spectrum handoff delay. Assuming each node is equipped with multiple radios, maintaining network connectivity is achieved by scheduling links to perform spectrum handoff. While maintaining the network connectivity, SH-MLMH problem further aims at minimizing the Total Handoff Completion Time (THCT) by scheduling links to perform spectrum handoff. This speeds up the process that the network recovers its optimal transmission. THCT is defined as the time for all the links to finish spectrum handoff. Notice that THCT is different from the spectrum handoff delay because the communication is not interrupted but just sub-optimal during it. To the best of knowledge, we are the first to study the SH-MLMH problem. We make the following contributions in this paper. (1) We formally formulate the SH-MLMH problem, and prove that it is NP-hard. (2) We propose a centralized algorithm to solve the SH-MLMH problem. We prove that the centralized algorithm can achieve a logarithmic approximation ratio. (3) The centralized algorithm requires global information, and thus is not efficient for scalable networks. We propose a distributed algorithm requiring only local information to overcome this disadvantage. The simulation results show that our proposed algorithms not only improve the network throughput but also reduce total handoff completion time compared with spectrum handoff without coordination.

For the third issue, we study channel assignment and bandwidth allocation in multi-radio multi-channel wireless network. Compared with existing works, this work has two main differences. First, existing works do not consider the impact of traffic load to the channel transmission quality. Here, traffic load refers to total bandwidth allocated to a channel. They assume that transmission quality of a channel is independent of the channel's traffic load. However, as described in the "Tragedy of the Commons" of Bandwidth Sharing game (TCBS) in [Nisan et al., 2005], the quality of a single channel will deteriorate as the increase of assigned bandwidth to that channel. In this work, we consider the relationship between transmission quality and traffic load. Second, existing works assumed that each node pair always has packets to send, implying saturated packet arrival rates, which is an extreme case in practice. The packet arrival rate is often limited in the real situation. In this work, we assume that the packet arrival rate is limited. We formulate this problem as a joint allocation problem, called Non-cooperative Joint Channel and Bandwidth Allocation problem, referred as NJCBA problem. We make following main contributions while solving the NJCBA problem. 1) We model the NJCBA problem as a non-cooperative static game. 2) We mathematically prove that there exist pure NEs for the NJCBA game. 3) We design two distributed algorithms to enable players to converge to the NEs quickly.

1.6 Impacts of Our study

We believe that our research on cross-layer optimization has three significant impacts. First, it will promote the development of wireless network. It is widely accepted that the layered network structure is not applicable for the wireless network, and the future wireless network requires more flexible network structure. However, it is still unknown how the layered network structure should be transformed to the new network structure. The cross-layer optimization method provides a practical way to transform the layered network structure to a more flexible one.

Second, more new applications of wireless network can be built based on our study. The layered network structure leads to sub-optimal performance of the wireless network. With cross-layer optimization, we improve the performance of wireless networks, such as higher throughput, better QoS support, and lower handoff delay. With improvement of network performance, new applications can be found. For example, the multi-hop wireless network can not provide enough bandwidth for online video streaming. With our study, this application can be supported since cross-layer optimization can improve the throughput.

Third, our research will improve the quality of services provided by wireless networks. We know that the wireless communication has changed people's life. It makes people's life become more convenient and more comfortable. For example, people can make phone call or view website through mobile phone. However, the performance of this service is not good enough. With cross-layer optimization, the performance of this service can be significantly improved. For example, in the past, people can only watch on-line video with low quality. With cross-layer optimization, people can enjoy on-line video with better quality. We believe that the cross-layer optimization will improve the quality of services provided by wireless networks.

1.7 Organization of the Dissertation

We describe the organization of this dissertation as follows.

- **Chapter 2 (Literature Review):** This chapter introduces the wireless network models that we will study. Specifically speaking, we introduce knowledge about cooperative communication technology, cognitive radio technology and multi-radio multi-channel technology. We also introduce some concepts about game theory.

- **Chapter 3 (Cooperative QoS Routing):** In this chapter, we design the first Cooperative QoS routing protocol (CQ-routing) for multi-hop wireless network. We first propose a cross-layer protocol stack, including a routing protocol, a MAC protocol and interfaces among them. The routing protocol is responsible to select both routing path and helper nodes. The MAC protocol is responsible to support cooperative transmission and collect link information. The routing protocol includes a routing algorithm and message mechanism. The routing algorithm selects the routing path with maximum available bandwidth by finding the routing path, corresponding helper nodes, and scheduling scheme. We implement the CQ-routing protocol in the real test bed, and evaluate them with experiments. The CQ-routing protocol can be divided into two parts

- **Chapter 4 (Multi-link Spectrum Handoff):** In this chapter, we study the multi-link spectrum handoff problem in multi-hop cognitive networks. We formulate the SH-MLMH problem as an optimization problem, which targets at minimizing the THCT while keeping the network connectivity. We propose both centralized and distributed algorithms to solve SH-MLMH problem. These two algorithms are responsible to decide the order of multiple links' spectrum handoff. It will trigger the handoff-aware routing protocol to perform routing update. We evaluate this protocol through simulation.

- **Chapter 5 (Non-cooperative Channel and Bandwidth Allocation):** In this chapter, we describe the works about Non-cooperative Joint Channel and Bandwidth Allocation problem (NJCBA). We model the NJCBA problem as a non-cooperative static game, called NJCBA game. In this game, each wireless node is selfish, and competing for limited resource to maximize its own benefit. We prove that there exists pure Nash Equilibrium for the NJCBA game. We design two distributed algorithms to enable players to converge to the NE point. We evaluate these two algorithms through simulations.
- **Chapter 6 (Conclusion and Future Works):** We summarize the research works and conclude this dissertation. We also discuss the future works and point out how to further optimize the network performance in the future.

Chapter 2

Literature Review

In this chapter, we investigate existing works about cross-layer optimization in next-generation wireless networks. We first investigate general methodologies of cross-layer optimization. Then we investigate cross-layer optimization in three kinds of wireless networks, including cooperative wireless networks, cognitive wireless networks, and multi-radio multi-channel wireless networks.

2.1 Classification of Cross-layer Optimization

In this subsection, we investigate existing works about cross-layer optimization. We classify existing works in this area according to layers that are involved, which is shown in 2.1.

- **Interaction among physical layer and upper layers:** The physical layer has parameters of transmit power, bit-error rate and coding/modulation. These parameters can be adjusted by upper layers to improve the network performance. [Aghvami and T.Le, 2001] proposed that software in the application layer may tune the physical layer parameters to improve the throughput. [Ebert and Wolisz, 1999] discussed the relationship between parameters in MAC layer, like packet length and the parameters in physical layer. Their results show that when this optimal transmission power is proportional to the packet length, minimum energy is consumed for sending a packet. Moreover, they show that varying the packet length according to the BER also helps reduce energy consumption. Existing works [Sharma et al., 2010], [Zhang and Zhang, 2008] study the problem of how to select relay nodes for cooperative communication in the physical layer and routing path in the network layer to improve the network performance.

- **Interaction among MAC layer and upper layers:** MAC protocol has parameter of forward error correction (FEC) and Automatic Repeat reQuest (ARQ). [Xylomenos and Polyzos, 2001] proposed to adapt parameters in the MAC layer to satisfy application layer's QoS requirement. The idea is based on the multi-service link layer for QoS which adapts the link layer services based on the traffic class. For example, frames of applications with a low delay requirement may be transmitted on priority. [DeSimone et al., 2003] find that retransmissions at the link layer may result in delays, which lead to TCP retransmissions and thus reduce throughput. [DeSimone et al., 2003] proposed that TCP and link layer could exchange retransmission information to avoid TCP retransmissions. [J.S.Wu et al., 2001, Sanmateu et al., 2002] proposed to utilize link layer hand-off information to reduce the hand-off latency for Mobile-IP.

- **Interaction among MAC layer and low layers:** Based on the information in the physical layer, the error control mechanisms at the link layer may be adapted to reduce the transmission errors. [Lettieri and Srivastava, 1998] proposed to adapt the parameter of Maximum Transmission Unit (MTU) for a particular BER to improve goodput and transmission range. [Lettieri and Srivastava, 1998] proposed to increase the frame length to improve the throughput according to the radio conditions.

- **Interaction among network layer and upper layers:** Because of the hand-off delay caused by Mobile-IP, TCP retransmission time-out (RTO) and back-off mechanism may reduce throughput. [Caceres and Iftode, 2002] proposed that the event of Mobile-IP hand-off should be sent to TCP protocols to reduce the retransmission latency.

- **Interaction among transport layer and upper layers:** TCP protocol may adapt its parameters to satisfy application's QoS requirement according to different application type. A user may assign priorities to the running applications. For example, the application with higher priority would indicate the need for higher download bandwidth. [Raisinghani and Iyer, 2003, Raisinghani et al., 2002] proposed that TCP may map the higher priority of an application to a larger receive window. In this way, application with higher priority can achieve higher download bandwidth.

- **Interaction among application layer and lower layers:** For multi-media application, information about channel conditions in the physical layer can be used to adapt the coding to achieve better performance. [Alwan et al., 2002, Liu and Zarki, 2003, Noble et al., 1997] proposed to adapt coding scheme according to the channel condition and available bandwidth. [Alwan et al., 2002, Liu and Zarki, 2003] also proposed similar methods. Information about coding scheme can also be used to save energy. [Alwan et al., 2002] proposed that information about the type of coding used by a video application could be used to discard some frames at the network interface to save power.

2.2 Existing Works about Cooperative Communication

In this section, we investigate existing works about cooperative communication and QoS routing. We classify existing works about cooperative communication into three categories according to different layers they belong to. Existing works about cooperative communication in physical layer focuses on designing cooperative schemes. [Sendonaris et al., 2003] proposed a decode-and-forward cooperative scheme, in which cooperative node detects the source node's packets and then retransmits the detected packets. In [Nosratinia et al., 2004], the performance of different cooperative schemes is discussed. Work in [Laneman et al., 2004] analyzes the capacity of decode-and-forward cooperative scheme using information theory. Another category of works about cooperative communication in physical layer study the problem of how to select helper nodes.

There are two categories of works about cooperative communication in the MAC layer. One category of related works in MAC layer focus on designing cooperative

Table 2.1: Existing Works about Cross-layer Optimization Methodologies

layer/layer	Physical layer	MAC layer	Network layer	Transport layer	Application layer
Physical layer	–	[Ebert and Wolisz, 1999]	[Sharma et al., 2010, Zhang and Zhang, 2008]	–	[Aghvami and T.Le, 2001]
MAC layer	[Lettieri and Srivastava, 1998, Ludwig et al., 2002]	–	[J.S.Wu et al., 2001, Sanmateu et al., 2002]	[DeSimone et al., 2003, Methfessel and Dombrowski, 2002]	[Xylomenos and Polyzos, 2001]
Network layer	[Sharma et al., 2010, Zhang and Zhang, 2008]	[J.S.Wu et al., 2001, Sanmateu et al., 2002]	–	[Caceres and Iftode, 2002]	–
Transport layer	–	[DeSimone et al., 2003, Methfessel and Dombrowski, 2002]	[Caceres and Iftode, 2002]	–	[Raisinghani and Iyer, 2003, Raisinghani et al., 2002]
Application layer	[Agrawal et al., 1998, Alwan et al., 2002, Ebert and Wolisz, 1999, Kravets and Krishnan, 2000, Liu and Zarki, 2003, Noble et al., 1997]	[Xylomenos and Polyzos, 2001]	–	[Raisinghani and Iyer, 2003, Raisinghani et al., 2002]	–

MAC protocol. [Liu et al., 2007] designed and implemented a cooperative MAC protocol based on 802.11 open source driver. Its experiment results show that cooperation among stations in a wireless LAN (WLAN) can achieve both higher throughput and lower interference. Work in [Zhang et al., 2010] sets up a testbed to evaluate the performance of cooperative communication in both physical layer and MAC layer. Its cooperative MAC protocols is based on TDMA MAC protocol. Authors in [Chen et al., 2008] study the problem of how to schedule transmission in single-hop wireless network to satisfy users' QoS requirements by exploring multi-user diversity. Another category of related works in MAC layer also discussed the problem of how to select helper nodes in single-hop wireless network. Authors in [Shi et al., 2008] proposes an optimal algorithm to assign helper nodes among multiple source-destination pairs. Authors in [Mukherjee and Kwon, 2010] proposes a distributed helper node selection scheme based on auction-theoretic strategies. Authors in [Phan et al., 2010] propose algorithms to select helper nodes for links between mobile users and base stations such that each user satisfies its quality-of-service (QoS) data rate while minimizing the energy consumption.

In network layer, existing works about cooperative communication study how to jointly select routing path and helper nodes to improve the network performance. Existing works in the network layer can be further divided into two categories according to their objectives. One category of existing works about cooperative routing focuses on saving energy. Authors in [Li et al., 2006] proposed a cooperative routing algorithm to find the routing path with the minimum energy cost from a source node to a destination node. Authors in [Sikora et al., 2006] studied cooperative routing in linear multi-hop wireless networks. They proposed routing algorithms to find a routing path which achieves desired end-to-end rate with minimum transmission power. Authors in [Pandana et al., 2006] proposed cooperative routing algorithm to maximize the network lifetime by saving energy in sensor networks. Authors in [Luo et al., 2006a] proposed three cooperative routing algorithms, namely, relay-by-flooding, relay-assisted routing, and relay-enhanced routing. These three cooperative routing algorithms study the tradeoff between achieved rate and required power. Works [Li et al., 2006, Luo et al., 2006a, Pandana et al., 2006, Sikora et al., 2006] all adopt the method of first finding a shortest path route and then find helper nodes to improve the route using cooperative communication, so these works can not fully explore the advantages of cooperative communication. Authors in [Ibrahim et al., 2008] propose algorithms to jointly find routing paths and helper nodes to satisfy certain bandwidth requirement and minimize consumed energy. Works [Ibrahim et al., 2008, Li et al., 2006, Luo et al., 2006a, Pandana et al., 2006, Sikora et al., 2006] only consider the single-source single-destination scenario, Zhang et al. in [Zhang and Zhang, 2008] proposed an algorithm to find cooperative routing path with minimum energy in multi-source multi-destination scenario.

The second category of existing works about cooperative routing consider the problem of utilizing cooperative to provide QoS support. Sharma et al. in [Sharma et al., 2010] study the problem of how to jointly select routing path and helper nodes to maximize the minimum bandwidth of multiple flows. Authors in [Ibrahim et al., 2008], [Sikora et al., 2006] also consider the problem finding the cooperative routing path, which satisfy certain bandwidth requirement. However, this work does not consider the interference and scheduling issues. When considering the interference effect, neighboring links can not work simultaneously, so their time slots must be carefully

Table 2.2: Existing Works about Cooperative Communication

layer	Objectives	Interference	Main contributions	Existing works
Physical layer	Improve throughput	No	Design cooperative scheme	[Laneman et al., 2004, Nosratinia et al., 2004, Sendonaris et al., 2003]
MAC layer	Improve throughput	No	Design MAC protocols	[Liu et al., 2007, Zhang et al., 2010]
MAC layer	Satisfy QoS requirement	Yes	Design scheduling algorithms	[Chen et al., 2008]
MAC layer	Improve throughput	No	Select helper nodes in single-hop networks	[Mukherjee and Kwon, 2010, Phan et al., 2010, Shi et al., 2008]
Network layer	Save energy	No	Design cooperative routing algorithms	[Lin et al., 2008, Maham et al., 2008, Sharma et al., 2010, Shi et al., 2008, Zhang and Zhang, 2008]
Network layer	Save energy	Yes	Design cooperative routing algorithms	[Zhang and Zhang, 2008]
Network layer	Provide QoS support	No	Design cooperative routing algorithms	[Ibrahim et al., 2008, Sharma et al., 2010, Sikora et al., 2006]

scheduled to achieve maximum throughput. The problem of estimate available bandwidth becomes very challenging. Zhang et al. in [Zhang and Zhang, 2008] proposes a concept of virtual graph to analyze the interference relationship in cooperative multi-hop wireless network. However, they did not consider the problem of finding routing path with enough bandwidth. Instead, they target at finding cooperative routing path with minimum energy. Therefore, no existing works have ever studied the problem of designing cooperative QoS routing to satisfy users' bandwidth requirement under the constraint of interference. Moreover, all existing works about cooperative routing evaluated the performance of their algorithms by simulations while we implement the CQ-routing protocol in the real testbed and evaluate its performance by experiments. We summarize existing works in table 2.2.

We also investigate related works about the problem of estimating available bandwidth in traditional multi-hop wireless networks. In [Zhu and M.Corson, 2002], [Zhai and Fang, 2006], the authors study the problem of how to find routing path with maximum available bandwidth in a traditional multi-hop wireless network. Chen et al. in [Chen et al., 2010] study the problem of finding routing path with maximum available bandwidth in a multi-rate and multi-hop wireless network. Gupta et al in [Gupta and Wu, 2009] performed an experimental comparison study of how to estimate available bandwidth for a routing path. However, all these works did not consider the cooperative communication technology. In conclusion, no existing works consider designing cooperative QoS routing with the interference constraint, so this is the first work about interference-aware cooperative QoS routing.

2.3 Existing Works about Cognitive Networks

In this section, we investigate existing works about cognitive networks, including works about spectrum handoff, spectrum selection and routing. Existing works about spectrum handoff are concerned with designing spectrum sensing algorithms in the physical layer to reduce the spectrum handoff delay of single link in the single-hop wireless networks. Spectrum sensing is the process for secondary users to perform spectrum scanning to find new available spectrums. We classify existing works about spectrum sensing algorithms into two categories, in-band spectrum sensing and out-of-band spectrum sensing.

In-band sensing adopts the periodic sensing structure in which secondary users perform spectrum sensing during the transmission period. The in-band sensing has the advantage of reducing handoff delay. However, it also has the disadvantage of reducing the throughput because spectrum sensing occupies the time in transmission period. Existing works focused on designing in-band sensing algorithm to minimize sensing time while satisfying different constraints. In [Ghasemi and Sousa, 2007, H.Kim and K.G.Shin, 2008, Kim and Shin, 2008, Lee and Akyildiz, 2008, Wang et al., 2007], the authors studied the issue of reducing sensing time to maximize the channel efficiency while maintaining the required detection probability. In [Jung and Liu, 2008, Pei et al., 2007], the authors studied the issue of reducing sensing time to maximize the throughput of the cognitive network while keeping the packet collision probability for the primary network under a certain threshold. In [Jia et al., 2008], the authors studied the tradeoff between the spectrum quality and spectrum sensing time.

The out-of-band spectrum sensing performs spectrum sensing when its current operating spectrums are reclaimed by the primary users. Compared with in-band sensing, the out-of-band spectrum sensing can improve the throughput of secondary users because it does not perform spectrum sensing during transmission time. However, it has the disadvantage of causing long handoff delay. Existing works proposed many approaches to minimize out-of-band sensing delay. Authors in [L.Luo and S.Roy, 2007] proposed an n-step serial search scheme based on correlated occupancy channel models, where the availability of current spectrum is assumed to be dependent on that of its adjacent spectrum bands. Authors in [Tian and Bi, 2006, Wang and Chen, 2008, Willkomm et al., 2005, Zhu et al., 2007] proposed that the secondary users can utilize redundant spectrums to reduce the handoff delay. They compare their approaches with pre-determined approaches. In the pre-determined approach, secondary users define a spectrum list in advance. The numerical results show that the out-of-band sensing approach outperforms the pre-determined approach.

Existing works about cognitive networks also study joint optimization of spectrum selection and routing. However, they do not consider the impact of spectrum handoff on these two issues. We classify existing woks about these two issues into two categories, deterministic and opportunistic approaches. Researchers in [Ma et al., 2008, Wang and Zheng, 2006, Xin, 2005] proposed deterministic approaches, in which routing and the operating spectrum are jointly determined for each hop in the routing path. Their results reveal that the deterministic approach outperforms a separated approach where routes and spectrums are selected independently. The authors in [Zhu et al., 2008] studied the joint optimization problem in the mesh network, and proposed a tree-structure based routing and spectrum selection algorithm for the backbone nodes of the mesh networks. The authors in [Khalife et al., 2008, Lei et al., 2008, Shiang and Schaar, 2008, Xin et al., 2008] proposed opportunistic solutions, which probabilistically estimates the available capacity of every spectrum, select routing path and spectrum in a probabilistic method.

Table 2.3: Existing Works about Cognitive Netowrk

layer	Issues	Methods	Existing works
Physical layer	Spectrum Handoff	In-band spectrum sensing	[Ghasemi and Sousa, 2007, H.Kim and K.G.Shin, 2008, Jia et al., 2008, Jung and Liu, 2008, Kim and Shin, 2008, Lee and Akyildiz, 2008, Pei et al., 2007, Wang et al., 2007]
Physical layer	Spectrum Handoff	Out-band spectrum sensing	[Ghasemi and Sousa, 2007, H.Kim and K.G.Shin, 2008, Jia et al., 2008, Jung and Liu, 2008, Kim and Shin, 2008, Lee and Akyildiz, 2008, Pei et al., 2007, Wang et al., 2007]
Physical layer	Spectrum Selection	Deterministic approach	[Ma et al., 2008, Wang and Zheng, 2006, Xin, 2005]
Physical layer	Spectrum Selection	Opportunistic approach	[Khalife et al., 2008, Lei et al., 2008, Shiang and Schaar, 2008, Xin et al., 2008]
Network layer	Routing algorithm	Deterministic approach	[Ma et al., 2008, Wang and Zheng, 2006, Xin, 2005]
Network layer	Routing algorithm	Opportunistic approach	[Khalife et al., 2008, Lei et al., 2008, Shiang and Schaar, 2008, Xin et al., 2008]

Existing work only studied the spectrum handoff problem in single-hop scenario. We study the spectrum handoff problem in a more general scenario that multiple links perform spectrum handoff in multi-hop cognitive networks. In the multi-hop scenario, we can keep the network connectivity to prevent the communication of switching links from being interrupted during spectrum handoff. We also try to minimize THCT such that the network can recover to stable state in minimum time. The spectrum handoff problem in the multi-hop scenario is more general and more challenging than that in the single-hop scenario.

2.4 Existing Works about Multi-Radio Multi-Channel

We investigate existing works about multi-radio multi-channel technology, and classify them as shown in table .2.4. We first classify existing works into cooperative channel assignment and non-cooperative one. In the first case, we assume that each wireless node will cooperate to optimize the network performance while in the second case, we assume that each wireless node competes for bandwidth resource to optimize its own performance rather than the network performance. Then we classify existing works into non-traffic-load-aware channel assignment and traffic-load-aware one. In the first case, we assume that each node always has packets to send, it consumes as much bandwidth as it can acquire while in the second case, we assume that wireless nodes have different traffic demands, and it will not consume more bandwidth than its traffic demand.

As we can see from table .2.4, for cooperative channel allocation problem, existing works considers both traffic-load-aware case, including [Kodialam and Nandagopal, 2005, Narayanan and Shende, 2001, Raniwala et al., 2004b, Raniwala and T.Chiueh, 2005, Shin et al., 2006] and non-traffic-load-aware case, including [Das et al., 2005, Mishra et al., 2005, Narayanan and Shende, 2001, van den Heuvel et al., 1998, Zheng and L.Cao, 2005]. Existing works about non-cooperative channel allocation problem include [Felegyhazi et al., 2007, Gao and Wang, 2008, Wu et al., 2008] and dynamic case. However, existing works does not consider impact of traffic load when studying channel allocation problem. From table .2.4, we can find that no existing works have been done about non-cooperative traffic-load-aware MRMC channel allocation problem, which is studied by this paper. Moreover, because we assume that traffic-load-aware case, we also need to consider how to allocate bandwidth among selected channels for each node. Therefore, this is the first paper to study the non-cooperative channel and bandwidth allocation problem using game theory approach.

Table 2.4: Existing Works about MRMC Channel Allocation

Behavior/Traffic-load	Cooperative channel allocation	Non-cooperative channel allocation
Non-traffic-load-aware	[Das et al., 2005, Mishra et al., 2005, Narayanan and Shende, 2001, van den Heuvel et al., 1998, Zheng and L.Cao, 2005]	[Felegyhazi et al., 2007, Gao and Wang, 2008, Wu et al., 2008]
Traffic-load-aware	[Kodialam and Nandagopal, 2005, Narayanan and Shende, 2001, Raniwala et al., 2004b, Raniwala and T.Chiueh, 2005, Shin et al., 2006]	————

Chapter 3

Interference-Aware Cooperative QoS Routing

3.1 Introduction

The cooperative communication is a new communication technology, which can improve poor links' transmission capacity by retransmission of helper nodes. However, most existing works about cooperative communication focus on exploring the advantages of cooperative communication in the physical layer. The impact of cooperative communication to the upper layer, like QoS routing, has not been fully understood yet. The routing algorithms, which selects both routing path and helper nodes in cooperative multi-hop wireless networks, are known as cooperative routing algorithms. There are two limitations for existing works about cooperative routing. First, existing works about cooperative QoS routing does not consider the interference effect among links, which significantly complicates the design of CQ-routing protocol. Although authors in [Zhang and Zhang, 2008] propose a concept of virtual graph to analyze the interference relationship in cooperative multi-hop wireless network, they did not consider designing cooperative QoS routing to satisfy users' bandwidth requirement. Second, all existing works about cooperative routing only evaluate their algorithms through simulations. It is widely accepted that simulations can not truthfully reflect communication protocol's performance, which can only be captured by real-world testbed. This paper sets up the first real-world cooperative multi-hop testbed to evaluate the performance of cooperative communication in the multi-hop scenario. To solve these two limitations, this paper targets at designing and implementing an interference-aware Cooperative QoS routing protocol (CQ-routing) in the HAWK testbed to provide QoS support.

We design the CQ-routing protocol based on existing routing protocols. There are two categories of routing protocols in multi-hop wireless network, proactive and reactive routing protocol. In proactive routing protocol, like OLSR, each wireless node periodically exchanges link information with other nodes, and always maintains routing paths to all other neighbors. In reactive routing protocol, like AODV, one wireless node only finds a routing path to a destination node when it needs to transmit packets to that node. The proactive routing protocol has many disadvantages. For example, it brings large overheads. What is more important, the proactive routing protocol is not applicable to provide QoS support because it does not consider single flows' QoS requirement. Therefore, in this paper, we design the CQ-routing protocol based on the AODV routing protocol. Because traffic flows usually come one by one in practical wireless networks and the CQ-routing protocol is on-demand, the CQ-routing protocol individually calculates cooperative routing path for each flow to satisfy its bandwidth requirement. We also consider a more general scenario in which multiple flows simultaneously find routing path. We adopt a method of resource reservation to coordinate multiple flows to select routing paths.

For each flow, there might exist multiple cooperative routing paths satisfying users' bandwidth demand. We adopt the routing metric, available bandwidth, to select one cooperative routing path. Given a routing path, its available bandwidth [Zhu and M.Corson, 2002] is defined as the maximum throughput that can be achieved along this path without interfering any ongoing flow in the network. Among these cooperative routing paths satisfying users' bandwidth demand, the CQ-routing protocol selects the cooperative routing path with maximum available bandwidth. This is because selecting routing path with large available bandwidth can balance the traffic load among different routing paths, and achieve less congestion. The available bandwidth metric is widely adopted in many existing QoS routing protocols [Chia-Cheng Hu, 2010]. They target at finding a routing path with maximum available bandwidth to satisfy users' bandwidth demand, which is denoted by MBRP problem.

We formulate the problem of finding a cooperative routing path with maximum available bandwidth as an optimization problem, denoted by Coop-routing problem. Given a cooperative multi-hop wireless network, a set of ongoing flows, and a new connection request from node s to t, the Coop-routing problem targets at finding a routing path P from node s to t, a set of helper nodes and a scheduling scheme for links in P such that P can achieve the maximum throughput and all ongoing flows are kept free from interference. It is easy to find that the MBRP problem is a special example of Coop-routing problem. We analyze and transform the Coop-routing problem as follows. We extend the concept of virtual graph in [Zhang and Zhang, 2008] to simplify the Coop-routing problem. In the virtual graph, virtual nodes and links are added to represent the links working in the cooperative transmission mode. We redefine the interference relationship, transmission capacity and available time for these virtual links. Based on the new virtual graph, we transform the Coop-routing problem to a simpler problem which targets at finding a single-path rouging with maximum throughput in the virtual graph. This single-path routing must satisfy a new constraint that a virtual node and its original node can not appear simultaneously in the same cooperative routing path. Otherwise, a cycle would happen in the cooperative routing path. This constraint is denoted by virtual cycle constraint. We define a conflict relationship between nodes which can not appear in the same routing path. Existing work in [Chia-Cheng Hu, 2010] proves that the MBRP problem is an NP-hard problem. We mathematically prove that the Coop-routing problem is a strong NP-hard problem, which is harder than the problem discussed in [Chia-Cheng Hu, 2010].

Even after the Coop-routing problem has been transformed, existing solutions to the MBRP problem is still not applicable to solve the transformed Coop-routing problem because of following reasons. 1) Virtual cycle constraint: Existing solutions does not consider the relationship that nodes can not simultaneously appear in the same routing path. 2) New interference relationship: When calculating available bandwidth, we must consider interference relationship to schedule transmissions. However, cooperative communication changes the interference relationship among links. In the traditional wireless network, two neighboring links interfere with each other because of broadcast nature of wireless medium. However, this assumption is not applicable for links working in the cooperative transmission mode. In the Coop-ABE problem, We need to calculate available bandwidth under the constraint of this new interference relationship. 3) More complicated capacity model: According to the Shannon capacity model, we find that the capacity of link working in direct transmission mode depends on its own Signal to Interference and Noise ratio (SINR) while the capacity of link working in cooperative transmission mode depends not only on its own SINR but also on other links' SINR. Therefore, we also need to consider this new capacity model when studying the Coop-routing problem. What is more important, existing solutions to the MBRP problem are not good enough. Existing solutions [Chia-Cheng Hu, 2010, DU and YANG, 2010] are all heuristic algorithms, and none of them can provide guaranteed-performance.

Because of above reasons, we propose new centralized and distributed algorithms to solve the Coop-routing problem. Hereafter, these two algorithms are referred to as centralized and distributed Coop-routing algorithms. It is hard to propose algorithms with guaranteed performance to solve Coop-routing problem. We consider a practical constraint that two wireless nodes can not be placed too close. We assume that the distance between neighboring nodes can not be less than a constant distance d_0, which is claimed in $\Omega(1)$ model in [Kuhn et al., 2003]. While considering this new constraint, we design a centralized Coop-routing algorithm which can achieve a constant approximation ratio. This property guarantees that the cooperative routing path found by the Coop-routing algorithm can achieve the available bandwidth, which is no less than a constant ratio of that found by the optimal solution. Because the MBRP problem is a special example of Coop-routing problem, the centralized Coop-routing algorithm is also applicable to solve the MBRP problem. To the best of our knowledge, this is the first approximation algorithm for Coop-routing problem with constant approximation ratio to the MBRP problem. However, the centralized Coop-routing algorithm requires global information, so it is not practical in the real wireless networks. We propose the distributed Coop-routing algorithm, to overcome the disadvantage of the centralized Coop-routing algorithm. The distributed Coop-routing algorithm requires only local information, and can be combined with the well-known AODV routing protocol.

Based on the distributed Coop-routing algorithm, we propose a cross-layer protocol stack for CQ-routing protocol, called Cooperative QoS Communication system (CQC). The CQC system includes a CQ-routing protocol, a Cooperative QoS MAC (CQ-MAC) protocol, and interfaces between these two protocols. The CQ-routing protocol is responsible to calculate selected routing path and helper nodes, and send this information to the CQ-MAC protocol through defined interface. We design the CQ-routing protocol by modifying AODV routing protocol. The distributed Coop-routing algorithm will work as a part of the CQ-routing protocol. Moreover, although the distributed CQ-routing algorithm only selects cooperative routing path for single flow, we also consider a more general scenario that multiple flows need to select cooperative routing paths. In the new scenario, we design a resource reservation mechanism to coordinate cooperative routing of multiple flows. The CQ-MAC protocol is responsible to collect link information, and support cooperative transmission. We design the CQ-MAC protocol by modifying 802.11 MAC protocol. In CQ-MAC protocol, the destination node can receive two copies of data transmitted from the source node and the helper node. This mitigates the channel fading effect. Finally, we implement the CQC system in our HAWK mesh testbed [Cao et al., 2009]. We evaluate the performance of the CQ-routing protocol by experiments. The results show that CQ-routing protocol can significantly improve network performance, in terms of available bandwidth and number of admitted flows. We summarize main contributions of this paper as follows.

- Prove that the Coop-routing problem is a strong NP-hard problem.
- Propose a centralized approximation algorithm with a constant approximation ratio to solve the Coop-routing problem.
- Propose a distributed algorithm to solve the Coop-routing problem, which requires only local information.
- Design and implement a cross-layer cooperative QoS routing protocol in the HAWK mesh testbed.

The rest of this paper is organized as follows. In section II, related works in this area are investigated. In section III, we describe system model, formulate the Coop-routing problem, and prove that it is a strong NP-hard problem. In section IV, both centralized and distributed algorithms are proposed to solve the Coop-routing problem. In section V, we design and implement the CQ-routing protocol in the HAWK mesh testbed. Section VI evaluates the performance of CQC system by experiments. Section VII concludes this paper.

3.2 System Model and Problem Analysis

3.2.1 System Model

In this paper, we consider a cooperative multi-hop wireless network. We make following assumptions in this paper. We assume that all links work in the same

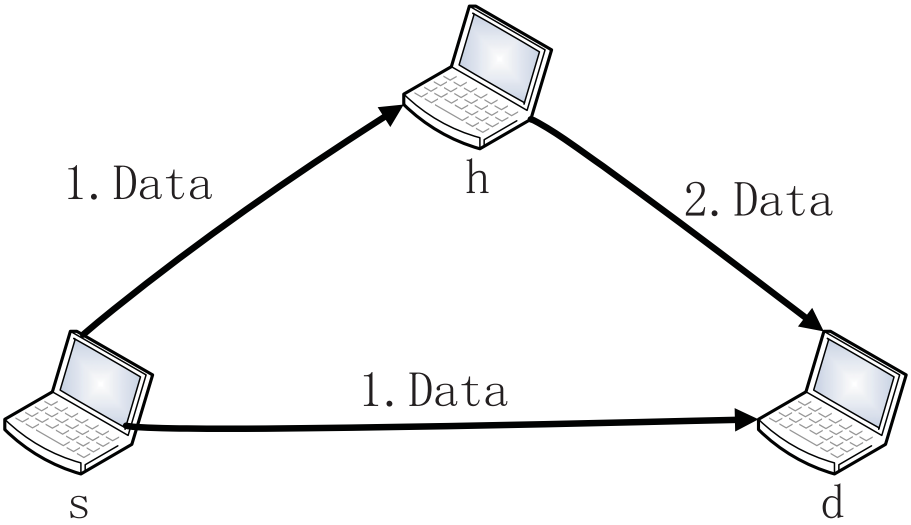


Fig. 3.1: DF cooperation scheme

channel, and an 802.11 Distributed Coordination Function (DCF) MAC protocol is adopted to handle multiple access in the MAC layer. We adopt the $\Omega(1)$ model introduced in Kuhn et al. [2003] to bound the distance between node pairs: the distance between any two nodes may not fall below a constant d_0. This assumption is reasonable because in the real wireless network there are physical limitations on how close to each other two nodes can be placed. We model the network as an undirected graph $G(V, E)$, where V denotes a set of n nodes, and E denotes a set of m links. We assume that there exists a link $e(u, v) \in E$ between node u and v if $d(u, v) \leq R_c$, where $d(u, v)$ is the physical distance between nodes u and v, and R_c is the communication range.

We assume that a Decode-and-Forward cooperation scheme (DF) is adopted to improve poor links' transmission capacity. We use a frame/time slot based model in [Laneman et al., 2004] for analysis. We assume that there are two time slots in each frame. As we can see from Fig. 3.1, in the first time slot, source node

s broadcasts messages to nodes h and d, which is called broadcast phase. In the second time slot, relay node retransmits packets to the destination node d, which is called the cooperative transmission phase. We distinguish two types of relay nodes in the network based on their functions. We denote a relay node used for cooperative purpose by helper node and denote a relay node for multi-hop relaying in the network layer by forwarder. We assume that each link can only have one helper to help it retransmit packets, and each node can work either as a helper or as a forwarder only once in a cooperative routing path. Similar assumption is also made in [Sharma et al., 2010], [Shi et al., 2008]. It is widely accepted the Shannon capacity is the theoretical upper bound on the rate for reliable transmission, so we use it approximate the capacity of direct and cooperative transmission. The capacity for links working on DF mode under the two-time-slot structure is given by [Laneman et al., 2004].

$$C_{DF}(s,h,d) = W * I_{DF}(s,h,d) \tag{3.1}$$

$$\begin{aligned} I_{DF}(s,h,d) = \frac{1}{2} * \min\{&\log_2(1+SINR_{sh}), \\ &\log_2(1+SINR_{sd}+SINR_{hd})\} \end{aligned} \tag{3.2}$$

The capacity for direct transmission is described as follows.

$$C_{direct}(s,d) = W * \log_2(1+SINR_{sd}) \tag{3.3}$$

To simplify our analysis, we use the Additive White Gaussian Noise (AWGN) model in [Chafekar et al., 2008] to approximate one link's SINR.

$$SINR_{ij} = \frac{P}{d(l_{ij})^{\alpha} N_0 W} \tag{3.4}$$

Here, W is the bandwidth, N_0 is white noise level, α is path loss factor, $d(l_{ij})$ is the distance between node i and j. Because W, N_0, and α are all constants, each link's SINR and capacity are fixed.

There are two kinds of interference model, protocol interference model [Tang et al., 2008] and physical interference model [Gurashish Brar, 2006]. In this paper, we adopt the protocol interference model to describe interference relationship among links. We assume that each link takes bidirectional transmission, and link l_{uv} interferes with link l_{xy} if and only if any two end nodes of these two links are within each other's interference range, $d(u, x) \leq R_I$, or $d(u, y) \leq R_I$, or $d(x, v) \leq R_I$, or $d(y, v) \leq R_I$. Given a link l, we define a link set, called Interference Link Set (IS) to denote the set of links interfering with link l. Moreover, this interference relationship is not applicable for links working in the cooperative transmission mode. For example, in Fig.3.1, in the broadcast phase of cooperative transmission, node s can successfully transmit packets to two nodes h and d only when all links interfering with links l_{sh} and l_{sd} keep silent. Later, we will construct a virtual graph and redefine the interference relationship for links working in the cooperative transmission mode.

We assume that there is a set of existing flows in network $G(V, E)$, $F = \{f_1, f_2, ...\}$. We denote the time fraction of link l occupied by flow f by $TF(l, f)$. We define a metric, called Available Time Fraction (ATF) to denote the left available time of link l except the time fraction occupied by existing flows within link l's interference range, $ATF(l) = 1 - \sum_{e \in IS(l)} \sum_{f \in F} |TF(e, f)|$. We define a new kind of routing path, called cooperative routing path as follows.

Definition 1. (Cooperative routing path) Given a network $G(V, E)$, and a connection request (s, t), the cooperative routing path between (s, t) is defined as a sub-graph S in G, which includes a routing path P from node s to node t, and a set of helper nodes $P^c = \{h_0, h_1, ..., h_n\}$. Here, h_i denotes the helper node for link $l_{i,i+1}$ in P. Moreover, h_i can be null, which denotes that link $l_{i,i+1}$ transmits in the direct transmission mode. The length of S is defined as that of P, $|S| = |P|$.

We define binary variable $x(m, S)$ to denotes whether or not node m serves as a forwarder in cooperative routing path S. We also define binary variable $y(m, l_{ij}, S)$ to denote whether or not node m serves as a helper for link l_{ij} in cooperative routing path S.

3.2.2 Problem Formulation

We formally formulate Coop-routing problem as following optimization problem which targets at finding a cooperative routing path with maximum available bandwidth.

Definition2. (Coop-routing problem) Given a network $G(V, E)$, each link's $SINR$ and ATF, a set of existing flows, and a new connection request $\rho(s, t)$ with required bandwidth $RBW(\rho)$, the Coop-routing problem is to find a cooperative routing path P from s to t, and a scheduling scheme for links in P such that P can achieve the maximum available bandwidth and existing flows are kept free from interfere. The bandwidth requirement $RBW(\rho)$ is also satisfied. We mathematically describe the problem as follows.

Objective: Maximize$(BW(P))$

subject to:

1. Path bandwidth constraint:

$$BW\,(P) = C(l_{i,i+1}, h_i) * TF(l_{i,i+1}, h_i, \rho), i \in P \tag{3.5}$$

2. Flow conservation constraint:

$$\begin{aligned} \forall l_{i,i+1}, l_{j,j+1} \in P, C(l_{i,i+1}, h_i) * TF(l_{i,i+1}, h_i, \rho) \\ = C(l_{j,j+1}, h_j) * TF(l_{j,j+1}, h_j, \rho), j \in P \end{aligned} \tag{3.6}$$

3. Feasible scheduling constraint:

$$\forall l \in IS(P), \sum\nolimits_{e \in IS(l)} TF(e, \rho) \leq ATF(l) \tag{3.7}$$

4. Relay constraint: $\forall i, j, m \in V$,

$$\sum\nolimits_{m \in V} x(m, P) + \sum\nolimits_{l_{ij} \in P} y(m, l_{ij}, P) \leq 1 \tag{3.8}$$

5. QoS constraint:

$$BW(P) \geq RBW(\rho) \tag{3.9}$$

Here, $BW(P)$ denotes the available bandwidth of path P, $IS(P)$ denotes the set of links interfering with links in P, $IS(P) = \underset{l \in P}{\cup} IS(l)$. $C(l_{i,i+1}, h_i)$ denotes the capacity of link $l_{i,i+1}$ working in the cooperative transmission mode with node h_i as its helper. $TF(l_{i,i+1}, h_i, \rho)$ denotes the time fraction consumed by link $l_{i,i+1}$ and its helper node h_i when it works in the cooperative transmission mode with node h_i as its helper.

The binary variable $x(m, P)$ to denotes whether or not node m serves as a forwarder in cooperative routing path P. We also define binary variable $y(m, l_{ij}, P)$ to denote whether or not node m serves as a helper for link l_{ij} in cooperative routing path S. Variables $x(m, P)$ and $y(m, l_{ij}, P)$ need to be determined to select the cooperative routing path P. Variable $TF(l_{i,i+1}, h_i, \rho)$ also needs to be allocated to decide the available bandwidth of the cooperative routing path P. The first constraint defines the path available bandwidth. The second constraint denotes the flow conservation constraint. The third constraint denotes each node within interference range of links in P should not be interfered. The fourth constraint denotes that any node can act only once either as a forwarder or a helper in P. The fifth constraint denotes that the bandwidth requirement of the new traffic flow should be satisfied. If the solution to the Coop-routing problem finds available bandwidth larger than bandwidth requirement of flow ρ, $BW(P) \geq RBW(\rho)$, flow ρ will be accepted without causing interference to ongoing flows.

3.2.3 Transform Problem

In this subsection, we transform the Coop-routing problem to a simpler problem by constructing a virtual graph. Authors in [Zhang and Zhang, 2008] propose the concept of virtual graph to model the new interference relationship among links working in cooperative transmission mode. However, this model does not consider the capacity and ATF of virtual link, so we extend the virtual graph to consider these new issues. The detailed algorithm, that constructs virtual graph is described in Algorithm .1. Specifically speaking, we construct a directed weighted graph as the virtual graph to simplify our analysis. We show the transformation by an example in Fig.3.2, in which node 2 is a helper for link l_{13} and l_{56}. We construct a directed weighted virtual graph $G^{'}(V^{'}, E^{'})$ shown in the right graph of Fig. 3.2 by following steps.

1) Compare transmission capacity: We compare the capacity of link l working in the direct transmission mode with that of link l working in the cooperative transmission mode. If link l working in the cooperative transmission mode can achieve higher capacity than working in the direct transmission mode, we go to step 2. Otherwise, we finish the procedures. In Fig.3.2, we compare the capacity of links l_{13} and l_{56} working in cooperative transmission mode with that of links l_{13} and l_{56} working in direct transmission mode. We find that both links l_{13} and l_{56} can achieve higher capacity when they work in the cooperative transmission mode, so we go to step 2.

2) Add virtual nodes and virtual links: We add virtual nodes and virtual links to denote the potential helpers and links working in the cooperative transmission mode. For example, in Fig.3.2, node 2 is a potential helper for link l_{56}, we add a virtual node 8 to represent the helper node 2, and two virtual links l_{58} and l_{86} to denote links between node 5 and node 6 working in the cooperative transmission mode. We also add virtual node 7 and virtual links l_{17} and l_{73} since node 2 also works as helper node for links l_{13}. To simplify our presentation, we define the set of virtual links in $G^{'}$ as Virtual link Set of $G^{'}$, denoted by $VS(G^{'})$. We define the set of original links in $G^{'}$ as Original link Set, denoted by $OS(G^{'})$. Given a virtual link v, we also define a link set, $L(v)$, to denote the original links that virtual link v represents. For example, in Fig.3.2, $L(l_{17}) = \{l_{12}, l_{13}\}$.

3) Label virtual links: We label virtual links with three metrics, including capacity, ATF and hop-count. As described by three-node model in 3.1 in subsection

3.2.1, the cooperative transmission from node s to d through node h can achieve capacity $C_{DF}(s,h,d)$. It takes two slots for the cooperative transmission, including the broadcast and cooperative transmission phase. In example of Fig.3.2, one virtual link l_{17} represents two original links $l_{12}, l13$ working in the broadcast transmission phase, so virtual link l_{17} can only work successfully when these two original links $l_{12}, l13$ can simultaneously work in the broadcast phase. Therefore, we define ATF of virtual link l_{17} as minimum ATF of these two original links in the broadcast transmission phase, $ATF(l_{17}) = \min\{ATF(l_{12}), ATF(l_{13})\}$. Another virtual link l_{73} represents one original link l_{23} working in the cooperative transmission phase, so virtual link l_{73} can work successfully when original link l_{23} works successfully. Therefore, we define ATF of virtual link l_{73} as that of original link l_{23}, $ATF(l_{73}) = ATF(l_{23})$. Moreover, because these two virtual links takes two time slots to transmit packet from node 1 to 3, its transmission capacity is defined as twice of the capacity achieved by cooperative transmission from node 1 to 3 through node 2, $C(l_{17}) = C(l_{73}) = 2 * C_{DF}(1,2,3)$. In our example, it is easy to verify that virtual link l_{17} can achieve the same bandwidth as that of broadcast transmission phase, $C(l_{17}) * ATF(l_{17}) = 2 * C_{DF}(1,2,3) * \min\{ATF(l_{12}), ATF(l_{13})\}$. We can also verify that virtual link l_{73} can achieve the same bandwidth as that of cooperative transmission phase, $C(l_{73}) * ATF(l_{73}) = 2 * C_{DF}(1,2,3) * ATF(l_{23})$. Because the length of a cooperative routing path is equal to that of a routing path P, we set hop-count for original link as 1 and that for virtual link as 0.5.

4) ***Redefine interference relationship:*** We redefine the interference relationship in the virtual graph by redefining original and virtual links' interference link set (IS). We define IS of an original link l as that of original link l and virtual links which include original links interfering with link l. For example, in Fig.3.2, in original graph, we suppose that $IS(l_{13}) = \{l_{12}, l_{23}, l_{34}\}$. In virtual graph, we set $IS(l_{13}) = \{l_{12}, l_{23}, l_{34}, l_{17}, l_{73}\}$. The IS of a virtual link v is defined as set of links interfering with original links included in v, $IS(v) = \underset{l \in L(v)}{\cup} IS(l)$, $L(v)$ denotes original links included in v. For example, in Fig.3.2, $IS(l_{17}) = \{l_{12}, l_{23}, l_{34}, l_{13}, l_{73}\}$.

5) Define conflict relationship: Because each node can only serve once either as a helper or as a forwarder in a cooperative routing path, this causes a new problem, called duplicated relay node selection problem, which means that single node might be selected more than once in single cooperative routing path. For example, when we try to find a cooperative routing path from node 1 to 6, path $P: 1 \rightarrow 7 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 8 \rightarrow 6$ is not a feasible path because node 2 serves twice as helper in path P. To represent this conflict relationship, we define a Conflict link Set (CS) to denote the set of links which can not simultaneously appear in the same cooperative routing path. CS of original link l_{ij} is defined as the set of virtual links which selects node i or j as the helper. CS of virtual link v is defined as the set of original links which is represented by link v and other virtual links which has the same helper as link v except cooperative link pair of link v. For example, in Fig.3.2, $CS(l_{12}) = \{l_{17}, l_{73}, l_{58}, l_{86}\}, CS(l_{17}) = \{l_{12}, l_{13}, l_{23}, l_{25}, l_{26}, l_{56}, l_{58}, l_{86}\}$.

Based on concept of virtual graph, we transform Coop-routing problem to a simpler routing problem, which seeks a single-path routing P and a scheduling scheme S to achieve the maximum available bandwidth.

Definition 3. (Transformed Cooperative Routing problem, TCoop-routing) Given a directed weighted graph $G(V, E)$, in which each link $e \in E$ is associated with three metrics, namely capacity $C(e)$, hop-count $h(e)$, threshold $ATF(e)$, and two link

Algorithm 1: Virtual-graph Algorithm

```
Input: Original graph $G(V, E)$, each link's $SINR$, $ATF$
Output: Virtual graph $G^{'}(V^{'}, E^{'})$
for each original link $l_{sd} \in E$ do
    Calculate direct transmission capacity of link $l_{sd}$, $C_{direct}(s, d)$;
    for each neighboring node $h$ of $l_{sd} \in E$ do
        Calculate cooperative transmission capacity of link $l_{sd}$ with help of node $h$, $C_{DF}(s, h, d)$;
        if $C_{DF}(s, h, d) \geq C_{direct}(s, d)$ then
            Add virtual node $v$ between node $s$ and $d$;
            Add virtual link $l_{sv}$, $l_{vd}$ between nodes $s$ and $d$;
            Label virtual links with transmission capacity, $C(l_{sv}) = C(l_{vd}) = C_{DF}(s, h, d)$;
            Label virtual links with ATF, $ATF(l_{sv}) = min\{ATF(l_{sh}), ATF(l_{sd})\}$, $ATF(l_{vd}) = ATF(l_{hd})$;
for each original link $l_{sd} \in E$ do
    for each virtual link $l_{uv} \in E$ do
        if $l_{uv}$ includes original link that interferes with original link $l_{sd}$ then
            $IS(l_{sd}) = IS(l_{sd}) \bigcup l_{uv}$;
        if $l_{uv}$ includes original link $l_{sd}$ then
            $CS(l_{uv}) = CS(l_{uv}) \bigcup l_{sd}$;
            $CS(l_{sd}) = CS(l_{sd}) \bigcup l_{uv}$;
for each virtual link $l_{uv} \in E'$ do
    for each original and virtual link $l_{mn} \in E'$ do
        if virtual link $l_{uv}$ includes link that interfere with $l_{mn}$ then
            $IS(l_{uv}) = IS(l_{uv}) \bigcup l_{mn}$;
```

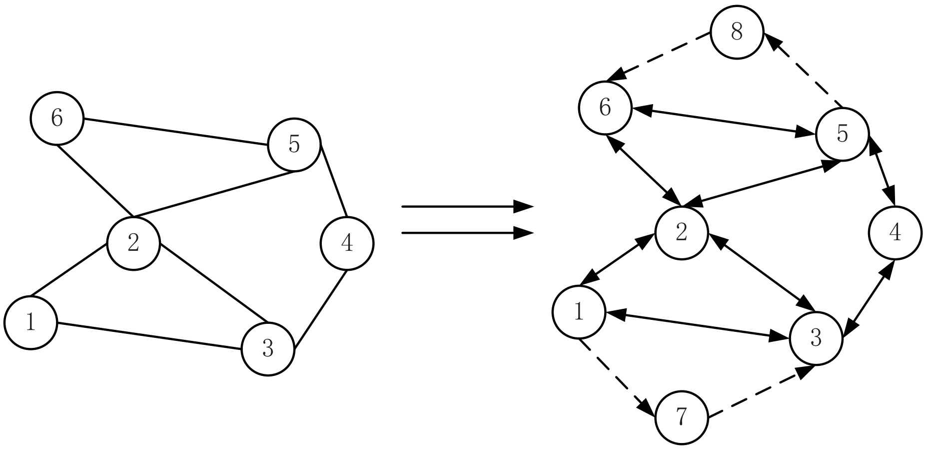


Fig. 3.2: Example of constructing virtual graph

sets, Interference-link-Set(e) ($IS(e)$) and a Conflict link Set $CS(E)$, a new connection request $\rho(s,t)$ with bandwidth requirement $RBW(\rho)$, our problem is to find a routing path P from s to t, and a scheduling scheme S for links in P such that P can achieve the maximum available bandwidth for connection $\rho(s,t)$ and existing flows are kept free from interference. We mathematically describe the problem as follows.

Objective: Maximize($BW(P)$),

subject to:

1. Path bandwidth constraint:

$$BW\,(P) = C(l_{i,i+1}) * TF(l_{i,i+1}, \rho) \tag{3.10}$$

2. Flow conservation constraint:

$$\forall l, l^{'} \in P, C(l) * TF(l, \rho) = C(l^{'}) * TF(l^{'}, \rho) \tag{3.11}$$

3. Feasible scheduling constraint:

$$\forall l \in IS(P), \sum_{e \in IS(l), e \in P} TF(e, \rho) \leq ATF(l) \tag{3.12}$$

4. Relay constraint:

$$\forall l \in E, x(l, P) + \sum_{e \in CS(l)} x(e, P) \leq 1 \tag{3.13}$$

5. QoS constraint:

$$BW(P) \geq RBW(\rho) \tag{3.14}$$

3.2.4 NP-Hard Proof

In this subsection, we prove that TCoop-routing problem is a strong NP-hard problem by reducing a strong NP-hard problem, Numerical Matching with Target Bounds (NMTB) problem to it. Because TCoop-routing problem is equivalent to Coop-routing problem, Coop-routing problem is also a strong NP-hard problem. The decision-version TCoop-routing problem is defined as follows: given an instance of TCoop-routing in definition 3 and a constant W, ask if there exists a routing path P between s and t, and a scheduling scheme for links in P, such that the available bandwidth of P is no less than W. We first define a well-known strong NP-hard problem, Numerical Matching with Target Sums (NMTS) as follows.

Definition 4. (Numerical Matching with Target Sums, NMTS problem) Instance: Disjoint sets X and Y, each containing m elements, a size $s(a) \in Z^+$ for each element $a \in X \cup Y$, and a target vector $< B_1, B_2, ..., B_m >$ with positive integer entries.

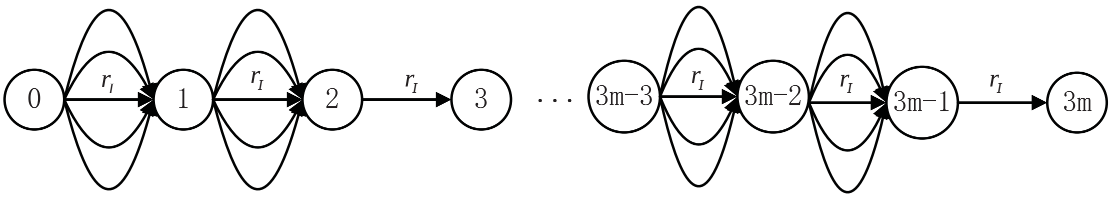


Fig. 3.3: Instance of Coop-routing

Question: Can $X \cup Y$ be partitioned into m disjoint sets $A_1, A_2, ..., A_m$, each containing exactly one element from each of X and Y, such that, for $1 \leq i \leq m$, $\sum_{a \in A_i} s(a) = B_i$?

We define a more general problem based on NMTS problem, called Numerical Matching with Target Bound(NMTB) as follows: Given an instance of NMTS, can $X \cup Y$ be partitioned into m disjoint sets $A_1, A_2, ..., A_m$, each containing exactly one element from each of X and Y, such that, for $1 \leq i \leq m$, $\sum_{a \in A_i} s(a) \leq B_i$?

It is easy to find that NMTS is a special example of NMTB problem. Because NMTS problem is a strong NP-hard problem, NMTB problem is also a strong NP-hard problem. Then we prove TCoop-routing problem is a strong NP-hard problem by reducing NMTB problem to TCoop-routing problem.

Theorem 1. TCoop-routing problem is a strong NP-hard problem.

Proof. Given a solution G to TCoop-routing problem, we can verify if G is a feasible solution or not within polynomial time, so TCoop-routing belongs to NP. Given an instance of NMTB problem as listed in definition 4, we construct an instance for TCoop-routing problem shown in Fig.3.3 as follows. We construct a virtual graph of $3m$ nodes. Specifically speaking, we set up a wireless network with linear network topology. We also set that the communication range and the interference range of each link is equal,$r_I = r_C$. Moreover, we set the distance between any neighboring node pair equal as the interference range, $d = r_C$. We add m links between node pairs $(3k, 3k+1)$ and node pair $(3k+1, 3k+2)$, $k \in N, 0 \leq k \leq m$. We denote the *ith* link between node m and n by $l^i_{m,n}$. We add one link between any node pair, $(3k+2, 3k+3)$, $k \in N, 0 \leq k \leq m$.

For links between $(3k+2, 3k+3)$, $k \in N, 0 \leq k \leq m$, we set $C(l^i_{3k+1,3k+2}) = C$, $ATF(l^i_{3k+1,3k+2}) = 2 * W/s_{min} + W/C, s_{\min} = \min_{a \in X \cup Y}\{s(a)\}$. According to the definition of IS, we can find IS of links between $(3k+2, 3k+3)$, $IS(l^i_{3k+2,3k+3}) = \{l^i_{3k+1,2k+2} or\ l^i_{3k+3,3k+4}, \forall i \leq m\}$. We set CS of link between $(3k+2, 3k+3)$ as null, $CS(l^i_{3k+2,3k+3}) = null$. For link $l_{0,1}$, we set $C(l^i_{0,1}) = W/s(x_i)$, $ATF(l^i_{0,1}) = B_0$. According to the definition of IS, we can find IS of link $l_{0,1}$, $IS(l^i_{0,1}) = \{l^i_{1,2}, \forall i \leq m\}$. We set CS of link $l^i_{0,1}$ as the *ith* link between any pair,$CS(l^i_{0,1}) = \{l^i_{n,n+1}, n \neq k\}$. For link $l_{3m-1,3m}$, we set $C(l^i_{3m-1,3m}) = W/s(y_i)$, $ATF(l^i_{3m-1,3m}) = B_m$. According to the definition of IS, we can find IS of link $l_{3m-1,3m}$, $IS(l^i_{3m-1,3m}) = \{l^i_{3m-2,3m-1}, \forall i \leq m\}$. We set CS of link $l^i_{3m-1,3m}$ as null, $CS(l^i_{3m-1,3m}) = null$. For links between $(3k, 3k+1), k \neq 0$, we set $C(l^i_{3k,3k+1}) = W/s(x_i)$, $ATF(l^i_{3k,3k+1}) = B_k + W/C$. According to the definition of IS, we can find IS of following link, $IS(l^i_{3k,3k+1}) = \{l^j_{3k,3k+1} or\ l^j_{3k+1,3k+2}, l_{3k-1,3k}, \forall j \leq m\}$. We set CS of link $l^i_{3k,3k+1}$ as the *ith* link between other node pairs $3n, 3n+1$ and $3n+1, 3n+2$,$n \in N, 0 \leq n \leq m$,$CS(l^i_{3k,3k+1}) = \{l^i_{3n,3n+1}, or l^i_{3n+1,3n+2}, n \neq k\}$. For links between $(3k+1, 3k+2), k \neq m$, we set $C(l^i_{3k+1,3k+2}) = W/s(y_i)$, $ATF(l^i_{3k+1,3k+2}) = B_k + W/C$. According to the definition of IS, we can find IS of following link, $IS(l^i_{3k+1,3k+2}) = \{l^j_{3k,3k+1} or\ l^j_{3k+1,3k+2}, l_{3k+2,3k+3}, \forall j \leq m\}$. We set CS of link $l^i_{3k+1,3k+2}$ as the *ith* link between other node pairs $3n, 3n+1$ and $3n+1, 3n+2$,$n \in N, 0 \leq n \leq m$,$CS(l^i_{3k+1,3k+2}) = \{l^i_{3n,3n+1}, or l^i_{3n+1,3n+2}, n \neq k\}$.

We prove that if there exists a solution G for NMTB problem, then there must exist a solution $G^{'}$ to TCoop-routing problem. We assume that there exists a partition $G = \{A_1, A_2, ..., A_m\}$ satisfying constraints in NMTB. We find a solution for TCoop-routing, including a routing path P, and a scheduling scheme S as follows. For each $A_k = \{x_i, y_j\} \in G$, we select two links $l^i_{3k,3k+1}$, $l^j_{3k+1,3k+2}$ for routing path P. We also select links $l^i_{3k+2,3k+3}$, $0 \leq k \geq m$ for path P. In this way, a routing path P between node 0 and 3m can be found. We allocate time slots to links $l^i_{3k,3k+1}$, $l^j_{3k+1,3k+2}$, $l^j_{3k+2,3k+3}$, $|TF(l^i_{3k,3k+1}, \rho)| = s(x_i)$, $|TF(l^j_{3k+1,3k+2}, \rho)| = s(y_j)$,$|TF(l^j_{3k+2,3k+3}, \rho)| = W/C$.

We prove that path P and scheduling scheme S satisfies constraints in TCoop-routing problem. It is easy to find that, $\forall l \in P, C(l) * TF(l, \rho) = W$, so $G^{'}$ satisfies the first constraint and the second constraint. For the third constraint, we prove that for any link l in this network, total time fraction occupied by transmissions in P is no more than link l's ATF. Without loss of generality , we pick up a link $l^m_{3k,3k+1}$ between $3k$ and $3k + 1$. Suppose that we select links $l^i_{3k,3k+1}$ and $l^j_{3k+1,3k+2}$ in P, $\sum\limits_{l \in IS(l^m_{3k,3k+1}), l \in P} |TF(l, \rho)| = |TF(l^i_{3k,3k+1}, \rho)| + |TF(l^j_{3k+1,3k+2}, \rho)| + |TF(l^j_{3k+2,3k+3}, \rho)| = s(x_i) + s(y_j) + W/C$. Because x_i and y_j are two selected elements in X and Y, $s(x_i) + s(y_j) + W/C \leq B_k + W/C$, $ATF(l^m_{3k,3k+1}) = B_k + W/C$, which means that the third constraint is satisfied. Using similar way, we can also prove that links $l^m_{3k+1,3k+2}$ and $l^m_{3k+2,3k+3}$ also satisfy the third constraint.

We prove that $G^{'}$ satisfies the fourth constraint by contradiction method. Suppose we select link $l^i_{3k,3k+1}$ in P, we prove that other links in $CS(l^i_{3k,3k+1})$ will not appear in P. We assume that there exists another link $l^i_{3k,3k+1}$ in $CS(l^i_{3k,3k+1})$ selected by path P. Because link $l^i_{3k,3k+1}$ appears in P, x_i is selected in A_k . Because link $l^i_{3n,3n+1}$ appears in P, x_i is selected in A_n. Therefore, $A_k \cap A_n = x_i \neq \emptyset$, this contradicts with assumption that A_k and A_n are disjoint from each other,$A_k \cap A_n = \emptyset$. So the fourth constraint is also satisfied, and is a feasible solution for TCoop-routing problem. Using similar method, we can also prove that if there exists a solution G' for TCoop-routing problem, there exists a solution G to NMTB problem. Therefore, TCoop-routing problem is a strong NP-hard problem. □

3.3 Algorithm Design

In this subsection, we propose both centralized and distributed algorithms to solve the Coop-routing problem. We prove that the centralized algorithm can achieve a constant approximation ratio. Because the centralized Coop-routing algorithm requires global network information, we propose a distributed Coop-routing algorithm, which requires only local information. The distributed Coop-routing algorithm is well combined with AODV routing protocol, and easy to implement.

3.3.1 Centralized Coop-routing Algorithm

We propose a centralized algorithm, called centralized Coop-routing algorithm to solve the Coop-routing problem. To guarantee that transmission on link l does not interfere with other links, link l cannot occupy time fraction more than ATF of links within its interference range. We define one metric, called Bottle-neck Available Time Fraction (BTF) to denote the available time fraction of link l while considering interference constraint, $BTF(l) = \min_{e \in IS(l)} ATF(e)$. We define another metric, called Bottle-neck Bandwidth (BBW) to denote the available bandwidth of link l when considering the interference constraint, $BBW(l) = \min_{e \in IS(l)} \{C(e) * BTF(e)\}$. Intuitively, we should select links with high BBW to achieve high available bandwidth. We construct a sub-graph, called reserved graph to select links with high BBW. We divide the centralized Coop-routing algorithm into three phases as follows.

1) Construct reserved graph: We first construct a virtual graph $G^{'}(V^{'}, E^{'})$ for the network G(V,E) according to rules in Sub-section III.C. To solve the duplicated relay node selection problem, we combine original node and its corresponding helper nodes as a supper node to represent that these links share a helper node. For example, nodes $2, 7, 8$ in the virtual graph of Fig.3.2 are combined as a super node 2 in the left graph in Fig.3.4. Because we need to select links with high BBW, we construct a sub-graph, called reserved graph, $G^{*}(V^{*}, E^{*}, mid)$ in the right graph of Fig.3.4, which only includes the links with BBW higher than the threshold mid. We use binary search method to update the threshold value mid until the shortest cooperative routing P from node s to t in the reserved graph G^{*} is found. We formally define reserved graph as follows.

Definition 5. (Reserved Graph) Given a weighted graph $G(V, E)$, and a threshold mid, Reserved Graph $G^{*}(V^{*}, E^{*}, mid)$ is defined as a sub-graph of G, which contains links with weights no less than mid,$w(e) \geq mid$.

2) Find the shortest path: We also prefer path with short length because the shortest path leads to less opportunity of interference. We modify Bellman Ford algorithm to find the shortest path in reserved graph $G^{*}(V^{*}, E^{*}, B)$. Moreover, we need to guarantee that virtual links are selected pair by pair. For example, we must select virtual link l_{12}^{virt},l_{23}^{virt} together rather than original link l_{12}^{orig} and virtual link l_{23}^{virt}. We extend Bellman Ford algorithm to satisfy this requirement. We differentiate two

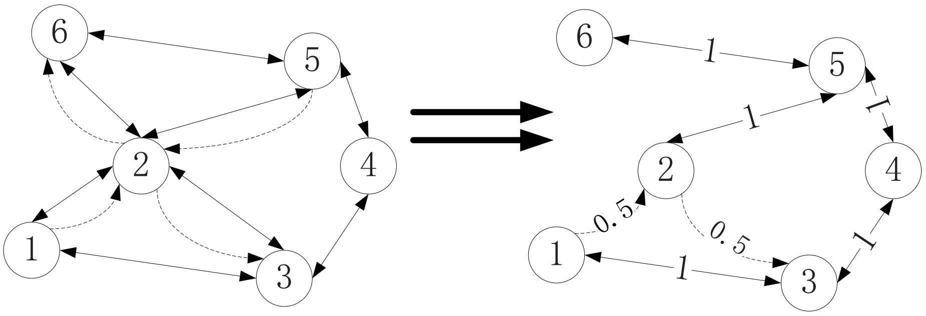


Fig. 3.4: Example of reserved graph

kinds of links, original link and virtual links. Each node needs to record both its forwarder (predecessor) and helper in previous hop. We treat the original link in the same way as original Bellman Ford algorithm. For virtual link, we treat them pair by pair. In each round, we calculate each node's distance to the source node by original link, and compare it with previous distance. Then we calculate each node's distance by two-hop virtual link pair. If this node's distance can be reduced, this node updates its distance, forwarders and helpers in previous hops. For example, in Fig.3.4, in a reserved virtual graph, we try to find the shortest path from node 1 to node 6. We select cooperative routing path: $1 \rightarrow 2(helper) \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6$ rather than path $1 \rightarrow 2 \rightarrow 5 \rightarrow 6$ because there does not exist original link between node pair $(1, 2)$.

3) Calculate available bandwidth In this phase, we calculate available bandwidth of the selected cooperative routing path P. We define a link set, called Path Interference Link Set (PILS) to denote the links interfering with link l in path P.

Definition 6. (Path Interference Link Set, PILS) Given a link l and a routing path P, PILS is defined as a set of links in P interfering with link l, $PILS(l, P) = \{e \in E | e \in P, e \in IS(l)\}$.

When considering interference from neighboring links in P, link l in path P can at most achieve following bandwidth, $BW(l) = BBW(l)/|PILS(l,P)|$. The available bandwidth of path P is determined by the bottle-neck link, $\min_{e \in P}(BBW(e)/PILS(e,P))$. We check whether the connection's bandwidth requirement is satisfied or not. If it is satisfied, we accept the new connection. Otherwise, the new connection is blocked. It is easy to verify that the time complexity of the centralized Coop-routing algorithm is polynomial. We prove that the centralized Coop-routing algorithm can achieve a constant approximation ratio.

Theorem 2. The centralized Coop-routing algorithm can achieve a constant approximation ratio of $g(R_I/R_C, R_I/d_0)$, which is a function of R_I/R_C and R_I/d_0.

Proof. We assume a cooperative routing path P^* generated by the optimal solution, and a cooperative routing path P generated by the centralized Coop-routing algorithm. We try to prove that $BW(P)/BW(P^*)$ is no less than a constant. We denote the minimum BBW of links in P^* by $BBW^*_{\min}$, and that of P by $BBW_{\min}$. Because we use binary search to select links for reserved graph, we find that $BBW_{\min} \geq BBW^*_{\min}/2$. Remember that $BW(P) = \min_{l \in P}(BBW(l)/|PILS(l,P)|)$. We prove that $|PILS(l,P)|$ is within a constant. Since $PILS(l,P) \subseteq IS(l), l \in P$, we prove $|IS(l)|$ is within a constant. We prove this by combining features of unit disc graph model and $\Omega(1)$ model. We prove that the area within interference range of link l is within a constant based on unit disc graph model. Based on $\Omega(1)$ model, two nodes cannot be close too much so we can prove that the number of nodes within interference range of link l is within a constant.

As mentioned before, IS of virtual link v include IS of original links included in v, so the maximum IS of a virtual link is obviously larger that of an original link. As we can see from Fig.3.5, node B works as the helper for link l_{AC}. We

Algorithm 2: Centralized Coop-routing

```
Input: Network G(V,E), each link's SINR, ATF, connection request $\rho(s,t)$
       and its required bandwidth $RBW$
Output: Cooperative Routing path P, a scheduling scheme S, BW(P)
Construct virtual graph $G^{'}(V^{'},E^{'})$;
Combine virtual node and its original node;
$mid = high = \max_{e \in E'}(BBW(e))$;
$low = \min_{e \in E'}(BBW(e))$;
while $high \geq low$ do
    Construct reserved graph $G^{*}(V^{*},E^{*},mid)$;
    for each node $v$ in $V^{*}$ do
        for each original link $l_{uv}$ adjacent to node $v$ do
            if $u.distance + uv.hops < v.distance$ then
                v.distance=u.distance + uv.hops;
                v.predecessor= u;
        for each virtual link pair $l_{mu}$,$l_{uv}$ adjacent to node $v$ do
            if $u.distance + uv.hops + mv.hops < v.distance$ then
                v.distance=u.distance + uv.hops+mu.hops;
                v.predecessor= m, v.helper=u;
    if $P \neq \phi$ then
        $BW(P) = \min_{e \in P}(BBW(l)/|PILS(l,P)|)$;
        $\forall e \in P, TF(e,\rho) = BW(P)/C(e)$;
        Break while ;
    else
        high= high/2,mid=(high+low)/2;
if $P = \phi$ or$BW(P) < RBW$ then
    Block the connection request
else
    Accept connection request $\rho$;
```

denote the virtual links between nodes A,B and B,C by dashed line. We can find that $|IS(l_{AB}^{virt})| = |IS(l_{AB})| \cup |IS(l_{AC})|$. We draw three circles around nodes A, B, C to denote the areas within their interference range. The maximum area within interference range of virtual link l_{AB}^{virt} can be achieved when the distance between A, B, C is equal to R_c. We calculate this maximum area with interference range of link l_{AB}^{virt} as follows. The total interference area is divided into four parts.

$$S = S_{\Delta DEF} + 3(S_{\widehat{DAF}} - S_{\Delta ADF}) \tag{3.15}$$

We calculate the degree of angle $\angle BAD$, denoted by θ .

$$\theta = 2\pi - 2\arccos(R_C/2R_I) - \pi/3 \tag{3.16}$$

We calculate length of link l_{DE} .

$$|DE| = 2R_I \sin(\theta/2) \tag{3.17}$$

We calculate the area of ΔDEF.

$$S_{\Delta DEF} = \frac{\sqrt{3}}{4}|DE|^2 = \frac{\sqrt{3}R_I^2 \sin^2(\theta/2)}{4} \tag{3.18}$$

We calculate the area of sector $\widehat{DAF}$.

$$S_{\widehat{DAF}} = \pi R_I^2 * \theta/2\pi = R_I^2 * \theta/2 \tag{3.19}$$

We calculate the area of ΔADF.

$$S_{\Delta ADF} = R_{_I}^2 * \sin(\theta)/2 \tag{3.20}$$

We calculate the totally interfered area.

$S = S_{\Delta DEF} + 3(S_{\widehat{DAF}} - S_{\Delta ADF})$

$$= R_{_I}^2(\tfrac{\sqrt{3}}{8} - \tfrac{\sqrt{3}}{8}\cos(\theta) + \tfrac{3}{2}\theta - \tfrac{3}{2}\sin\theta) = R_{_I}^2 f(\theta) \tag{3.21}$$

Because the distance between any node pair is no less than a constant d_0 , for any node u within interference range of $l_{_{AB}}^{virt}$, we draw a circle centered around node u with radius $d_0/2$. We use D to denote the set of such disks. The number of disks in D is upper-bounded by number of non-intersecting disks within interference range of $|D| \leq 4R_{_I}^2 f(\theta)/(\pi d_0^2)$. Because each node v only appears once in a cooperative routing path, the number of links in P within interference range of $l_{_{AB}}^{virt}$ is bounded by number of nodes interfering with link $l_{_{AB}}^{virt}$. Therefore, we can prove that $BW(P) = \min_{e \in P}(\frac{BBW(l)}{|PILS(l,P)|}) \geq \frac{\pi d_0^2 * BW(P^*)}{8R_{_I}^2 f(\theta)}, approx = \frac{BW(P)}{BW(P^*)} \geq \frac{\pi d_0^2}{8R_{_I}^2 f(\theta)}$. To simplify our presentation, we denote the approximation ratio $\frac{\pi d_0^2}{8R_{_I}^2 f(\theta)}$ by function $g(R_I/R_C, R_I/d_0)$. $\square$

3.3.2 Distributed Coop-routing Algorithm

We propose a polynomial distributed algorithm, called distributed Coop-routing algorithm to solve the Coop-routing problem. In the distributed Coop-routing algorithm, each node only needs to know link information within its interference range. Each node collects link information by exchanging messages with neighbors within its

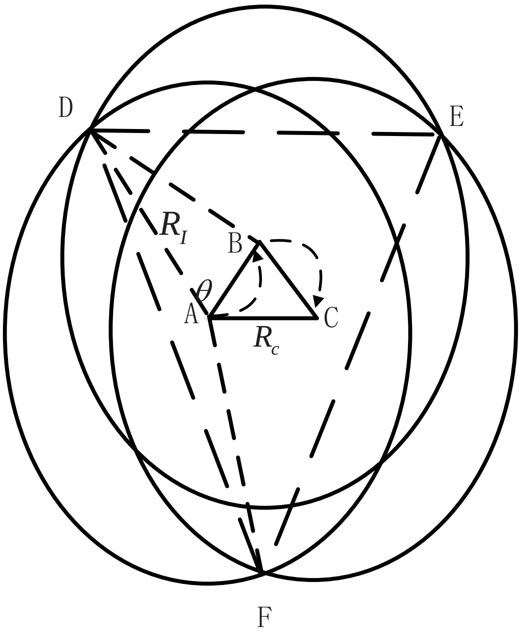


Fig. 3.5: Interference area of virtual link

interference range. In the real multi-hop wireless network, wireless node can not know the physical distance without support from special device, like GPS. Because wireless node does not know its physical position, we use hop-count to approximate interference range. We assume that two nodes within K hops (K=2 or 3) interfere with each other, and node pairs with larger hop distance will not interfere with each other. Similar approximation is widely adopted in many existing works, such as [Raniwala et al., 2004a]. The Coop-routing algorithm is well combined with the AODV routing protocol. The basic idea of distributed Coop-routing is to enable forwarders to select helpers when proceeding RREQ messages. We divide the distributed Coop-routing algorithm into three phases as follows.

1) Construct local virtual graph: Each node constructs a local virtual graph following the rules in Sub-section III.C. The local virtual graph of node n contains

Algorithm 3: Distritbuted Coop-routing

```
Input: Network $G(V,E)$, each link's $SINR$, $ATF$, connection request,$\rho(s,t)$,
       and its required bandwidth $RBW$
Output: Cooperative Routing path $P$, a scheduling scheme $S$, $BW(P)$
Each node exchanges link information with $K$-hop neighbors;
Each node constructs local virtual graph $G^{'}(V^{'},E^{'})$;
Source node $s$ broadcasts RREQ message;
Node $j$ receiving RREQ from node $i$ deletes helpers and forwarders selected by
previous links from local virtual graph;
Node $j$ calculates $EBW(P,e)$ metric for each original link or virtual link pair
$(i,j)$;
if there exists link e with $EBW(P,e) \geq BW(P)$ then
    Construct local reserved graph $G^{*}(V^{*},E^{*},BW(P))$;
    Select original link or virtual link with maximum capacity between node
    pair $(i,j)$;
else
    Select original link or virtual link pair with maximum EBW as forwarding
    links between $(i,j)$;
    $BW(P) = EBW(P,e)$;
if $BW(P) \geq RBW$ then
    Node j reserves resource for flow $\rho(s,t)$,
    $ATF(e) = ATF(e) - RBW/Cap(e)$;
    Node j forwards RREQ messages;
else
    Node j stops forwarding RREQ messages;
if node t receives multiple RREQs with $BW(P) \geq RBW$ then
    $t$ selects path $P$ with maximum $BW(P)$, and replies a RREP message;
else
    Block the connection requests;
if Node j receives RREP messages within timeout then
    Node $j$ sends a HELPER Message to the selected helper node $h$;
else
    Node $j$ releases reserved resources, $ATF(e) = ATF(e) + RBW/Cap(e)$;
```

only the local virtual nodes and links to represent the potential helpers and links working in cooperative transmission mode within its interference range.

2) ***Proceed RREQ:*** Node s broadcasts RREQ messages to find a cooperative routing path P from s to t, which satisfies its bandwidth requirement. We modify RREQ messages to carry more information, including the new connection's bandwidth requirement, RBW, $BW(P)$, selected forwarders and helpers. To avoid selecting duplicated relay node, node j receiving RREQ message deletes forwarders and helpers selected by previous forwarding links from local virtual graph. Node j receiving RREQ message from node i finds helpers for link l_{ij} by selecting original link or virtual link pair between node pair (i,j). We define a metric, called Expected path available BandWidth (EBW), to select forwarding links between nodes i and j.

Definition 7. (Expected path available BandWidth, EBW) Given a partial path $P = \{l_0, l_1, ..., l_n\}$ between nodes s and t, a link e, EBW(P,e) is defined as the path available bandwidth for the extended path $P^{'} = \{l_0, l_1, ..., l_n, e\}$, $EBW(P, e) = BW(P^{'})$.

$EBW(P, e)$ represents the expected available bandwidth of partial path $P^{'}$ which is the extended path by adding link e to partial path P, so we can use it as a metric to select forwarding link. We calculate $EBW(P, e)$ as follows. Because link e only interferes with links within its interference range, we calculate available bandwidth for links in $PILS(P^{'}, e)$, $BW(PILS(P^{'}, e)) = \min_{l \in PILS(P^{'}, e)}(BBW(l)/|PILS(P^{'}, l)|)$. We calculate $EBW(P, e) = min\{BW(P), BW(PILS(P^{'}, e))\}$. Because we need to select virtual link pair by pair, we define EBW metric for a partial path $P = \{l_0, l_1, ..., l_n\}$ and a virtual link pair $\{v_1, v_2\}$ as the available bandwidth for the extended path $P^{'} = \{l_0, l_1, ..., l_n, v_1, v_2\}$, $EBW(P, \{v_1, v_2\}) = BW(P^{'})$. We can use similar method as above to calculate $EBW(P, \{v_1, v_2\})$.

Node j calculates EBW metric for each original link and virtual link pair between node pair (i, j). Node j checks if there exists original link e or virtual link pairs between node pair (i, j) with $EBW(P, e)$ no less than $BW(P)$. If there exist multiple such forwarding links, we construct a local reserved virtual graph with $BW(P)$ as the threshold. We select forwarding links with maximum capacity because the link with larger capacity consumes less time to achieve $BW(P)$. In this way, it causes less interference for links in P. If there do not exist such forwarding links which can achieve $BW(P)$, we select the link with maximum EBW as the forwarding link. Node j adds IP address of selected helper and its own to RREQ messages, and forwards RREQ messages.

3) Proceed RREP: If the destination t receives multiple RREQ messages, it checks if $BW(P)$ is larger than the new connection's required bandwidth RBW. If $BW(P)$ is larger than RBW, destination node t select a path P with maximum available bandwidth, and replies an RREP message to the source node s along the reverse direction of path P. Otherwise, the new connection request will be blocked. The RREP message contains selected forwarders and helpers. If node j receives RREP message and it selects node h as the helper for the forwarding link l_{ij}, it sends a HELPER message to selected helper node h. Node h receiving HELPER message adopts the cooperative transmission mode to retransmit packets. The source node s receiving RREP message starts to transmit packets along path P.

4) Coordinate multiple flows: We extend the distributed Coop-routing algorithm to a more general scenario in which multiple flows select cooperative routing path. We design following resource reservation mechanism to coordinate multiple flows to select cooperative routing. When a forwarding node j receiving RREQ message, it checks if selected forwarding link e can provide enough bandwidth. If the forwarding link e can provide enough bandwidth, it will reserve such resource for this routing path by updating its ATF parameter. If node j does not receive $RREP$ message within timeout, node j releases such resource by updating its ATF parameter. Otherwise, this resource will be utilized by this routing path. By doing this, multiple flows can avoid occupying the same resource.

3.4 Design and Implementation of Cooperative QoS Communication System

3.4.1 Overview of CQC System

In this subsection, we provide an overview of the cross-layer Cooperative QoS Communication (CQC) system. As we can see from Fig.3.6, CQC system includes three components, namely a Cooperative QoS routing protocol (CQ-routing), a Cooperative QoS MAC protocol (CQ-MAC), interfaces between CQ-routing and CQ-MAC. We implement CQ-routing protocol based on open-source AODV -0.9.5 routing protocol, and implement CQ-MAC protocol based on open-source madwifi-9.3.2 MAC protocol. Moreover, CQ-MAC protocol is implemented in the kernel space of the Linux operating system while CQ-routing protocol is implemented in the user space of the Linux operating system. The kernel version of the Linux operating system is 2.6.31.14. We implement the interfaces between CQ-routing and CQ-MAC protocols using netlink sockets. The CQ-MAC protocol consists of three modules: NET80211 module, ATH module and HAL module. NET80211 module contains generic 802.11

functions and callback functions which can be overridden by devices. The ATH module defines Atheros specific callbacks for the net80211 module and accesses the hardware through the HAL module. HAL module is a short term for "Hardware Abstraction Layer", which refers to a piece of software that all direct access to the Atheros hardware is routed through. The CQ-routing protocol consists of two parts, distributed Coop-routing algorithm and message mechanism. Coop-routing algorithm is responsible to find the cooperative routing path. The message mechanism is used to periodically exchange link information among nodes within K hops.

The CQC system works as follows. 1) Each node collects link information, including SINR, ATF, through the ATH module of its CQ-MAC protocol. 2) The CQ-MAC protocol sends the collected link information to the CQ-routing protocol by netlink sockets. 3) The message mechanism of CQ-routing protocol exchanges link information among neighbors within K hops. 4) When a new connection request comes, the distributed Coop-routing algorithm of CQ-routing protocol calculates a cooperative routing path P. 5) The CQ-routing protocol informs the CQ-MAC protocol the selected cooperative routing path P. 6) If this node is selected as a helper or it needs to transmit packets to a helper, NET80211 module of the CQ-MAC protocol adopts the DF cooperative transmission mode to transmit packets. Otherwise, it adopts the direct transmission mode to transmit packets.

3.4.2 Implementation of CQ-routing Protocol

In this subsection, we describe how to implement CQ-routing protocol based on AODV routing protocol. As mentioned in subsection V.A, CQ-routing protocol has two main functionalities: exchange link information and find cooperative routing path.

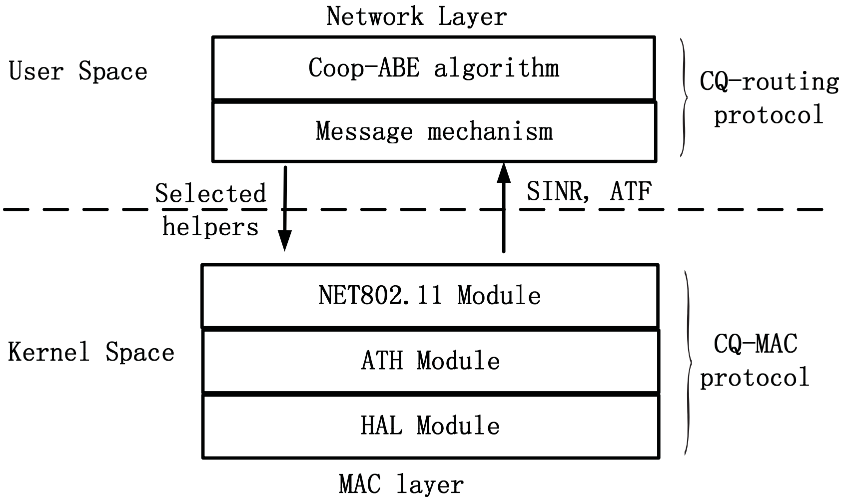


Fig. 3.6: Framework of Cooperative QoS Communication System

We first explain how to implement the first functionalities, exchange link information. As shown in Fig.3.6, each node collects the link information from the madwifi driver. Each node exchanges link information with neighboring nodes by HELLO and LINK messages. In the original AODV routing protocol, HELLO message is used to exchange message among one-hop neighbor to identify the connectivity relationship. We modify the HELLO message such that it can carry more link information, such as SINR and ATF. The format of new HELLO message is shown in Fig.3.8(b). The HELLO message is only exchanged among one-hop neighbor, so its hop-count is set to 0, and TTL value is set to 1. We also define a new kind of message, called LINK message. Each node's LINK message contains information about SINR, ATF of its neighboring links. LINK message is exchanged among neighbors within K hops. K denotes the interference range in term of number of hops. The LINK message has the same format as the HELLO message. Its hop-count value is set to K, and TTL value is set to K. In this way, each node can know link information about neighboring links within its interference range.

We explain how to find cooperative routing path as follows. As described in subsection IV.B, we modify RREQ, RREP message to find cooperative routing path. As shown in Fig.3.8(c), RREQ message contains following new entries. First, it contains the path available bandwidth calculated by previous forwarding links along the path. Second, it contains SINR information about previous forwarding links along the path. Third, it also contains the time fraction that the previous forwarding links requires to achieve the attached path available bandwidth. As shown in Fig.3.8(d), RREP message contains following new entries. First, it contains the path available bandwidth of selected cooperative routing path. Second, it contains not only the forwarders in the selected cooperative routing path but also the helpers in that path. We also define new HELPER message to inform selected helpers to retransmit packets for the forwarding links. The format of HELPER message is shown in Fig.3.8(a). Each node maintains a cooperative routing table to record the related routing information. The original routing table does not contain information about helper node. We modify this such that the helper node to reach the next hop is also included in that table. The format of the cooperative routing table is described in Fig.3.9. The exchange of control messages in Coop-routing protocol is shown in Fig.3.7. The source node s broadcasts RREQ message to find routing path to reach the destination node d. Each forwarder receiving RREQ message selects the helper node for the forwarding link according to the distributed Coop-routing algorithm in subsection IV.B. It also updates the path available bandwidth according to the distributed Coop-routing algorithm. The destination node selects a cooperative routing path with maximum available bandwidth, and sends back RREP message along the selected routing path. The selected forwarder sends a HELP message to the selected helpers to inform that

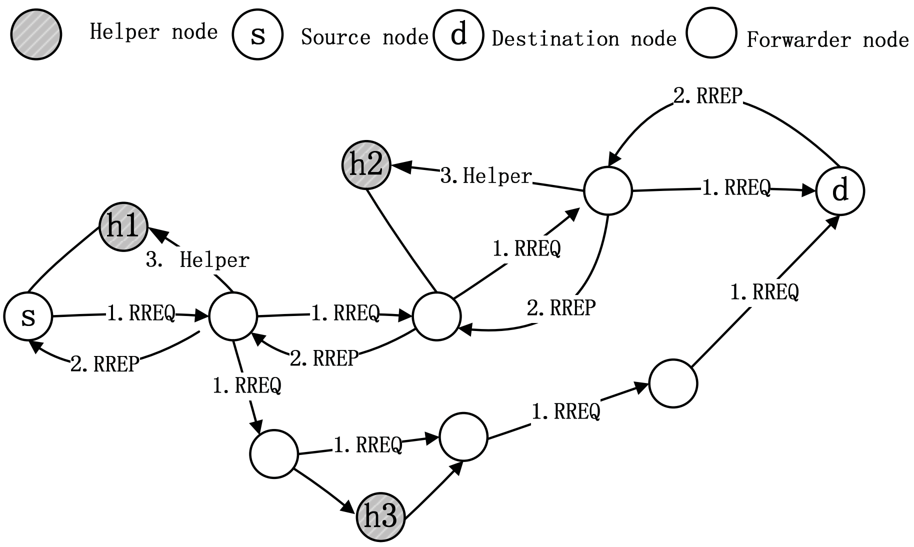


Fig. 3.7: Control message of CQ-routing protocol

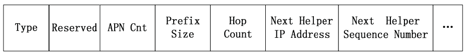
it is selected as HELPER. The helper will adopt the cooperative transmission mode to transmit packets.

3.4.3 Implementation of CQ-MAC Protocol

In this subsection, we describe how to implement CQ-MAC protocol. As mentioned before, CQ-MAC protocol has two main functionalities: collect link information and support DF cooperative transmission mode. We describe them as follows.

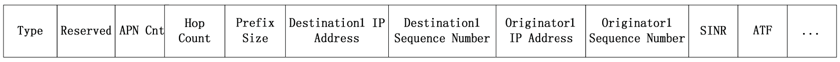
Collect link information

We modify madwifi drivers to collect link information, such as $SINR$, ATF, and transmission rate. For $SINR$ information, each node can directly read $SINR$ information from ATH module of madwifi driver. The transmission rate of neighboring



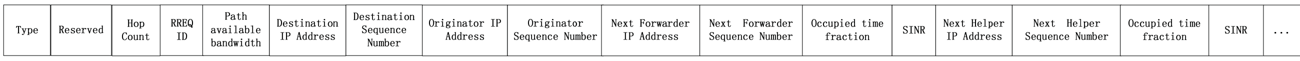
Type	Reserved	APN Cnt	Prefix Size	Hop Count	Next Helper IP Address	Next Helper Sequence Number	...

(a) format of HELPER message



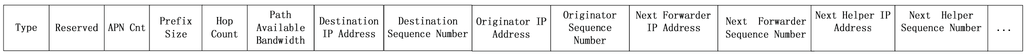
Type	Reserved	APN Cnt	Hop Count	Prefix Size	Destination1 IP Address	Destination1 Sequence Number	Originator1 IP Address	Originator1 Sequence Number	SINR	ATF	...

(b) format of HELLO and LINK message



Type	Reserved	Hop Count	RREQ ID	Path available bandwidth	Destination IP Address	Destination Sequence Number	Originator IP Address	Originator Sequence Number	Next Forwarder IP Address	Next Forwarder Sequence Number	Occupied time fraction	SINR	Next Helper IP Address	Next Helper Sequence Number	Occupied time fraction	SINR	...

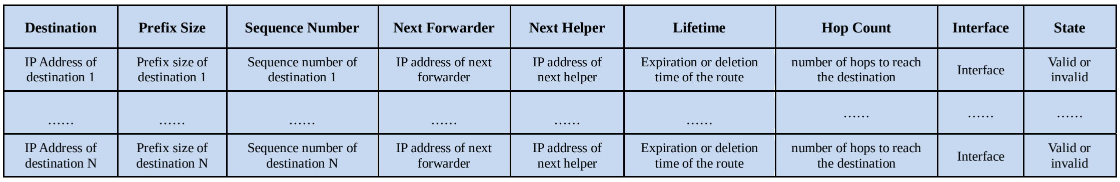
(c) format of RREQ message



Type	Reserved	APN Cnt	Prefix Size	Hop Count	Path Available Bandwidth	Destination IP Address	Destination Sequence Number	Originator IP Address	Originator Sequence Number	Next Forwarder IP Address	Next Forwarder Sequence Number	Next Helper IP Address	Next Helper Sequence Number	...

(d) format of RREP message

Fig. 3.8: format of control message

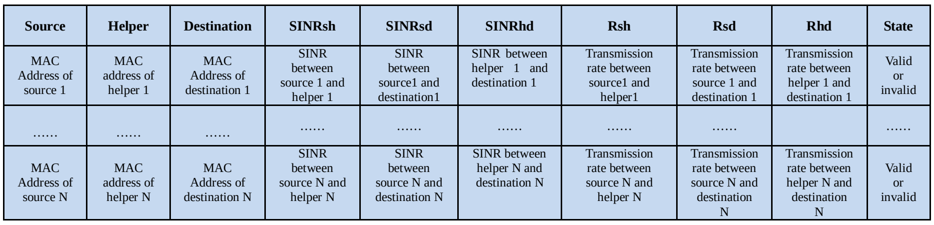


Destination	Prefix Size	Sequence Number	Next Forwarder	Next Helper	Lifetime	Hop Count	Interface	State
IP Address of destination 1	Prefix size of destination 1	Sequence number of destination 1	IP address of next forwarder	IP address of next helper	Expiration or deletion time of the route	number of hops to reach the destination	Interface	Valid or invalid
……	……	……	……	……	……	……	……	……
IP Address of destination N	Prefix size of destination N	Sequence number of destination N	IP address of next forwarder	IP address of next helper	Expiration or deletion time of the route	number of hops to reach the destination	Interface	Valid or invalid

Fig. 3.9: cooperative routing table

links can be read from MAC header of received packets. We estimate ATF information by passively listening to all ongoing transmissions. In 802.11 MAC protocol, each node senses the channel by CSMA mechanism [Luo et al., 2006b]. If the signal strength is larger than a threshold, the channel is assumed busy. We modify the driver such that this information is recorded periodically by the driver. According to the history information, we calculate the ratio of time that channel is assumed idle to the total time as ATF.

Support DF cooperative transmission mode

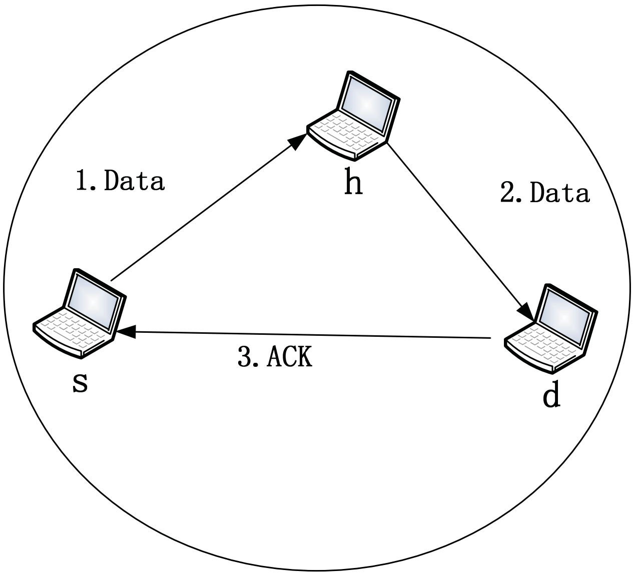


Source	Helper	Destination	SINRsh	SINRsd	SINRhd	Rsh	Rsd	Rhd	State
MAC Address of source 1	MAC address of helper 1	MAC Address of destination 1	SINR between source 1 and helper 1	SINR between source1 and destination1	SINR between helper 1 and destination 1	Transmission rate between source1 and helper1	Transmission rate between source 1 and destination 1	Transmission rate between helper 1 and destination 1	Valid or invalid
……	……	……	……	……	……	……	……		……
MAC Address of source N	MAC address of helper N	MAC Address of destination N	SINR between source N and helper N	SINR between source N and destination N	SINR between helper N and destination N	Transmission rate between source N and helper N	Transmission rate between source N and destination N	Transmission rate between helper N and destination N	Valid or invalid

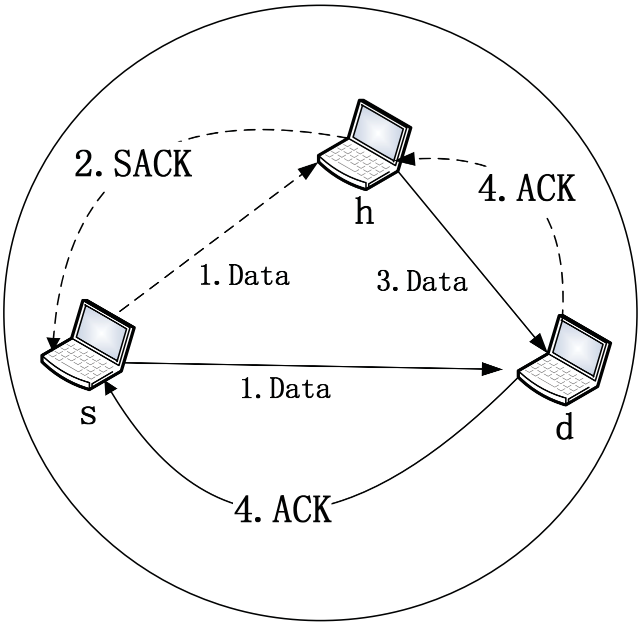
Fig. 3.10: cooperative MAC table

We design CQ-MAC protocol to support DF cooperative scheme by modifying an existing cooperative MAC protocol, Coopmac protocol in [Liu et al., 2007]. The Coopmac in [Liu et al., 2007] is also based on 802.11 MAC protocol. The Coopmac protocol has some limitations, which are solved by the CQ-MAC protocol. Due to space limitations, we only introduce the basic idea of CQ-MAC protocol and explain the difference between CQ-MAC and Coopmac protocol.

The basic idea of CQ-MAC protocol is shown in Fig.3.11(b). In the first slot, source node s broadcasts packet destined to node d, which is overheard by node h. The source node and helper node wait for ACK message from the destination node d. If within timeout, the helper node does not overhear the ACK message, the helper



(a) Data frame exchange of Coopmac protocol



(b) Data frame exchange of CQ-MAC protocol

Fig. 3.11: Comparison of Coopmac and CQ-MAC protocol

node h sends back a Spurious ACK message (SACK) to the source node s. The helper node h sets the destination d's MAC address as the source MAC address of this SACK message. The source node receiving SACK message assumes the destination node has received the data packet, and it will not retransmit this packet. After that, the helper node h contends for a time slot to retransmit the data packet to the destination node d until it overhears an ACK message from destination node d. If the destination node d receives the data packets, it sends back an ACK message. This ACK message is overheard by the helper node h, and the helper node h do nothing. Finally, the destination node d receives the data packet either from the source node s or from the helper node h. We summarize the difference between CQ-MAC and Coopmac protocol as follows.

- **Select helper node:** The Coopmac protocol selects the helper node in the MAC layer. The CQ-routing protocol selects the helper node in the network layer, and it informs the CQ-MAC protocol the information about selected helper nodes. Corresponding to the cooperative routing table in the network layer, the CQ-MAC protocol construct a cooperative MAC table in MAC layer. The format of the cooperative MAC table is shown in Fig.3.10. Using ARP protocol, the cooperative MAC table includes the MAC address of the source, helper and destination of neighboring links. It also includes SINR and transmission rate about neighboring links.

- **Retransmit packets:** We compare the retransmission mechanism of Coopmac and CQ-MAC shown in Fig .3.11. In Coopmac protocol, when the source node s broadcasts data packet, it sets the helper's MAC address as the destination MAC address. This causes a problem that the destination node d drops the packet which it received from the source node s, and it can only receive packet from the helper node h. In the CQ-MAC protocol, the source node s sends packets with destination node d's MAC address as the destination MAC address in the packet. In 802.11 MAC protocol, each node will drop all packets whose destination MAC address is not itself. We modify this mechanism, such that the helper node h can receive all packets. When the helper receives one packet, it decides whether it serves as the helper for this packet by checking the cooperative MAC table. In traditional 802.11 MAC protocol, a node forwards a packet by replacing the Source MAC address with its own MAC address, and the Destination MAC address with next-hop destination's MAC address. However, in CQ-MAC protocol, helper node h does not change the MAC header, and retransmits the packet as it received. The helper node h stops retransmitting the packet until it overhears ACK message from node d. In CQ-MAC protocol, the destination node d can receive data packet either from both source node s or from helper node h, so this improves the opportunity that the data packet is successfully received compared with the Coopmac protocol.

- **Handle ACK messages:** In the Coopmac protocol, the source node waits for the ACK message from the destination node d within a timeout. If the source node does not receive packets within timeout, the source node retransmits the packets. This causes a problem. If the source node does not receive the ACK message, it is possible that the helper node receives the packet and the destination node does not receive the data packet. In that case, if the source node retransmits the packet, it consumes more time than the helper node retransmits the packet. In the CQ-MAC protocol, we solve this problem by making the helper node generate a Spurious ACK message (SACK). The helper node h waits for the ACK message from the destination node h. If within timeout, ACK message is not received, the helper node h sends back a SACK message to the source node s. Please notice that the source MAC address of this SACK is the destination node d. The source node s receiving ACK message assumes the destination node has received the packet and stops retransmitting the packet. This improves the transmission efficiency compared with the coopmac protocol.

3.5 Performance Evaluation

3.5.1 Experiment Setup

In this section, we evaluate the performance of the CQ-routing protocol by both experiments and simulation. The evaluation metrics include available bandwidth, transmission delay, packet loss ratio, number of admitted flows, and routing discovery time. We compare the performance of CQ-Routing (CQR) protocol with three other routing protocols, the Shortest Non-Cooperative Routing protocol (SNCR), the Shortest Cooperative Routing protocol (SCR), the Widest Non-Cooperative Routing (WNCR) protocol. The SNCR routing protocol selects the routing path with minimum number of hops, and it did not select helper nodes. The SCR is quite similar to the cooperative routing protocol in [Lin et al., 2008]. It first selects a shortest routing path using AODV routing protocol. After that, each forwarding node selects helper node according to SINR values for links between helper node and source node and SINR for links between helper node and destination node. However, they did not consider the interference and cooperative capacity issue. The widest routing protocol considers interference and selects routing path with maximum available bandwidth, but it does not consider selecting helper nodes to utilize cooperative communication.

We do experiments to evaluate the performance of CQ-routing protocol in two different network scenarios, the indoor network scenario and the outdoor network scenario. In the indoor network scenario, we set up a regular linear network topology along the corridor, and analyze the behavior of different cooperative routing protocols. We compare the available bandwidth achieved by different cooperative routing protocols. In the outdoor network scenario, we set up an irregular linear network topology along the corridor, and compare the performance of different cooperative routing protocols, including available bandwidth, transmission delay, packet loss ratio, admission ratio, and routing discovery time. We use our T902 mesh router for the HAWK project [Cao et al., 2009] shown in Fig.3.12 to set up the wireless mesh network. Each T902 mesh router is equipped with three wireless network cards, and six antennas. In these three wireless network cards, one is used to connect mobile client, and the other two are used to connect other mesh routers to build up the backbone network. In our experiment, each node uses only one radio to build up the backbone network. Our CQ-routing protocols and other three routing protocols are installed in the T902 mesh router, and work in the backbone network to connect the mesh routers. All radios in the backbone network are set to work in channel 9 of 802.11b, and work with the same transmission power 18dbm.

There are two limitations for evaluation through experiments. First, we can only evaluate the performance of CQ-routing protocol with limited number of nodes in certain network topology. We did not evaluate the performance of CQ-routing protocol in large-scale network. Second, because the centralized CQ-routing algorithm

Fig. 3.12: T902 mesh router

requires global information, its performance is not evaluated in testbed. To overcome these disadvantages, we further evaluate the performance of both centralized and distributed CQ-routing algorithms in different network topologies through simulations. We evaluate the available bandwidth and the number of admitted flows in different scenarios. We analyze the impact of bandwidth demand, number of hops, and node density to the performance of CQ-routing protocols.

3.5.2 Experiments in indoor environment

In this subsection, we test the CQ-routing protocol in the indoor environment. We set up a linear network along a corridor shown in figure 3.13. We deliberately put node 1 in a room and open the door such that the link quality of link l_{13} is bad because the wall prevents the signal from node 1. However, node 1 and 3 can still receive each other's HELLO message, so they still assume they are one-hop neighbor. The quality of link l_{12} is good because the door is open and nodes 1 and 2 are visible to each other. Similar configuration is set between nodes $5, 6, 7$.

Fig. 3.13: Indoor network topology

Table 3.1: Parameters for cooperative routing

links	SINR (dB)	Bandwidth	ATF
l_{21}	23	4Mbps	0.84
l_{32}	32	3.9Mbps	0.433
l_{31}	5	596Kbps	0.38
l_{43}	40	6Mbps	0.56
l_{45}	31	7Mbps	0.56
l_{57}	8	1.3Mbps	0.8
l_{67}	25	2.1Mbps	0.221
l_{56}	27	1.6Mbps	0.12

To illustrate the behavior of different routing protocols, we trace and analyze cooperative routing from node 4 to 1 in TABLE 3.2. To facilitate our analysis, We test related parameters for each single link, and show them in TABLE 3.1. Using the SNCR protocol, node 4 sends packets to node 1 along the shortest routing path: $4- > 3- > 1$. We test the available bandwidth of this shortest routing path, and find that it achieve bandwidth of 183Kbps. Using the WNCR protocol, node 1 selects three-hop routing to node 4: $4- > 3- > 2- > 1$. The test shows that the routing path found by the WNCR protocol achieves bandwidth of 1.24Mbps. Compared with

Table 3.2: Available bandwidth of different routing path

Routing protocol	selected Paths	Available Bandwidth
$SNCR$	$4->3->1$	183Kbps
$WNCR$	$4->3->2->1$	1.11Mbps
SCR	$4->(3,2,1)$	1.45Mbps
CQR	$4->(3,2,1)$	1.45Mbps
$SNCR$	$4->5->7$	735Kps
$WNCR$	$4->5->7$	735Kps
SCR	$4->(5,6,7)$	471Kbps
CQR	$4->5->7$	735Kps

the shortest routing protocol, the widest routing protocol improves available by about 500%. This result shows the shortest path might achieve very litter bandwidth if the link quality is bad. The WNCR considers the interference issue, and it selects routing path with large available bandwidth.

Using SCR routing protocol, node 4 sends packets to node 1 along the shortest cooperative routing path: $4->(3,2,1)$. Here, (3,2,1) denotes node 3 sends packets to node 1 with node 2 as helper node. We can find that the SINR of link l_{31} is much weaker than that of links l_{32} and l_{21}. This verifies that SCR selects helper node according to the channel condition. We test the available bandwidth of the routing path found by SCR routing, and find that it achieve bandwidth of 1.35Mbps. Compared with the WNCR protocol, SCR routing protocol improves available bandwidth by about 30%. This result shows that the cooperative communication can improve poor link's transmission capacity compared with traditional multi-hop forwarding. Using the CQ-routing protocol, node 1 selects the same routing path as that of SCR routing.

To differentiate between the CQ-routing protocol and the SCR routing protocol, we analyze the cooperative routing from node 4 to node 7. We find that SCR routing protocol selects cooperative routing path, $4- > (5, 6, 7)$ while the CQ-routing protocol selects the shortest routing path, $4- > 5- > 7$. We analyze the reason for this phenomenon, and find that this is because the SCR routing protocol does not consider interference effect. As shown in 3.1, although link quality (SINR) of links l_{56} and l_{67} is larger than that of link l_{57}, their ATF is much less than that of link l_{57}. This shows that there exists more interference in the environment for links l_{56} and l_{67} than for link l_{57}. What is more important, the available bandwidth achieved by the shortest routing path achieves more available bandwidth than that of the cooperative routing selected by the SCR routing protocol. This reminds us that it is important to appropriately to selecting cooperative routing. Otherwise, the cooperative communication may even select cooperative routing path with smaller available bandwidth than traditional routing protocols.

To further evaluate the performance of CQ-routing protocol, we test available bandwidth of flows from node 1 to other 6 nodes using different routing protocols. The testing results are shown in Fig. 3.14. The SNCR protocol achieves the worst performance while CQ-routing protocol achieve the maximum available bandwidth. Compared with SNCR protocol, the CQ-routing protocol can improve the available bandwidth by about 302%. This is because the CQ-routing protocol selects routing path with metrics of available bandwidth while considering the interference and cooperative communication. Compared the WNCR protocol, the CQ-routing protocol can improve the available bandwidth by about 27%. This is because the CQ-routing protocol adopts the cooperative communication to improve poor links' transmission capacity, for example, it selects node 2 as the helper node for link l_{13}. Moreover, in

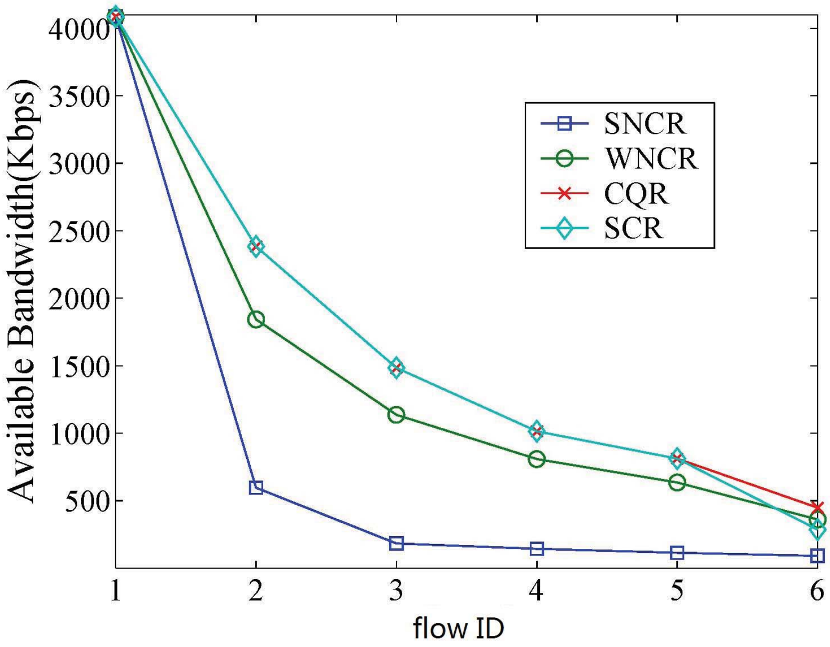


Fig. 3.14: Comparison of available bandwidth in indoor network

this simple linear network, the CQ-routing protocol achieves the same performance as the SCR protocol for most flows except the flow F_{17}. As we analyzed above, this is because for flow F_{17}, CQ-routing selects different cooperative routing path because it considers the interference and capacity issue. What is more important, the SCR protocol achieves less bandwidth than the WNCR protocol. This result shows that it is important to calculate available bandwidth while considering interference effect.

3.5.3 Experiment in outdoor environment

We set up a wireless mesh network of 19 nodes in the campus of Hong Kong Polytechnic University in Fig.3.15. The network topology is not regular, and the distance between two neighboring nodes is from 30m to 50m. We do following experiments to evaluate the performance of CQ-routing protocol.

Fig. 3.15: Network topology for experiment

Evaluation of Network Performance

In this subsection, we compare the performance of CQ-routing protocol with other three routing protocols, in terms of available bandwidth, transmission delay, and packet loss ratio. We randomly selects 10 source-destination node pairs, and use various routing algorithms to select routing paths for them. For each routing algorithms, we test available bandwidth of their selected routing paths by iperf tools. We test different routing protocols 20 times. We compare available bandwidth using different routing protocols in Fig.3.16. The result shows that the CQ-routing protocol achieves the maximum available bandwidth. The CQ-routing protocol can at most improve available bandwidth by 175.38%, 41.21%, 168.46% compared with the SNCR, WNCR, and SCR protocol. In some cases, the CQ-routing protocol achieve the same available bandwidth as that of SNCR and WNCR protocol. This shows that in that scenario, it is not appropriate to adopt cooperative transmission considering the link quality and interference condition. Moreover, the performance of SCR protocol fluctuates significantly. In some cases, the SCR routing protocol achieves less available

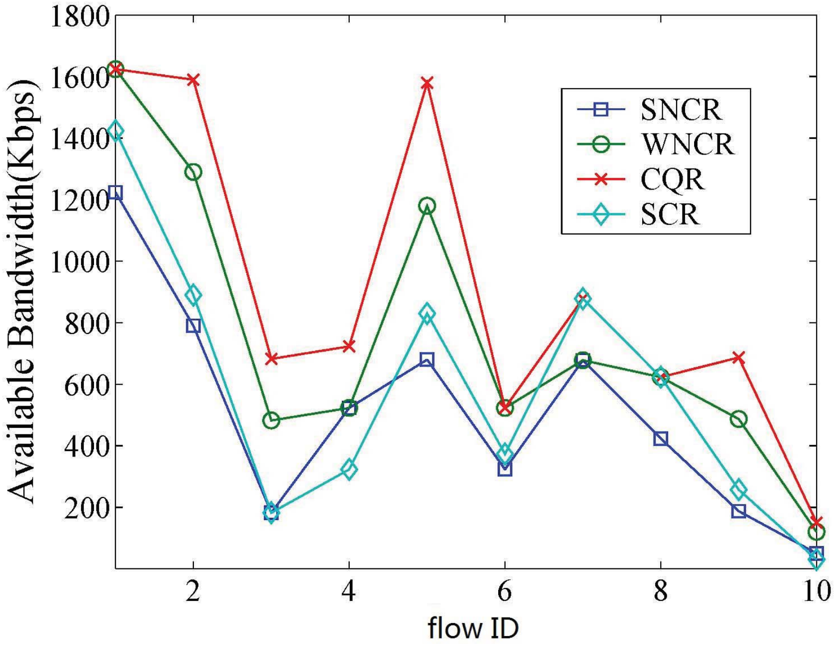


Fig. 3.16: Comparison of available bandwidth in outdoor network

bandwidth than the SNCR routing protocol. This shows that the if the helper node is not properly selected, the cooperative communication will even reduce the available bandwidth compared with the SNCR routing protocol.

We can also find similar results for transmission delay and packet loss ratio. We test transmission delay using different routing protocols in Fig.3.17. The result shows that the CQ-routing protocol at most reduces the transmission delay by 43% compared with the SNCR protocol. Although SNCR protocol selects routing with minimum number of hops, the link quality of the selected routing path leads to long transmission delay. The CQ-routing protocol at most reduces the transmission delay by 27% compared with the WNCR protocol, and 43% compared with the SCR protocol. The performance of SCR protocol fluctuates significantly. In some cases, it reduces the transmission delay while it enlarge the transmission delay in some

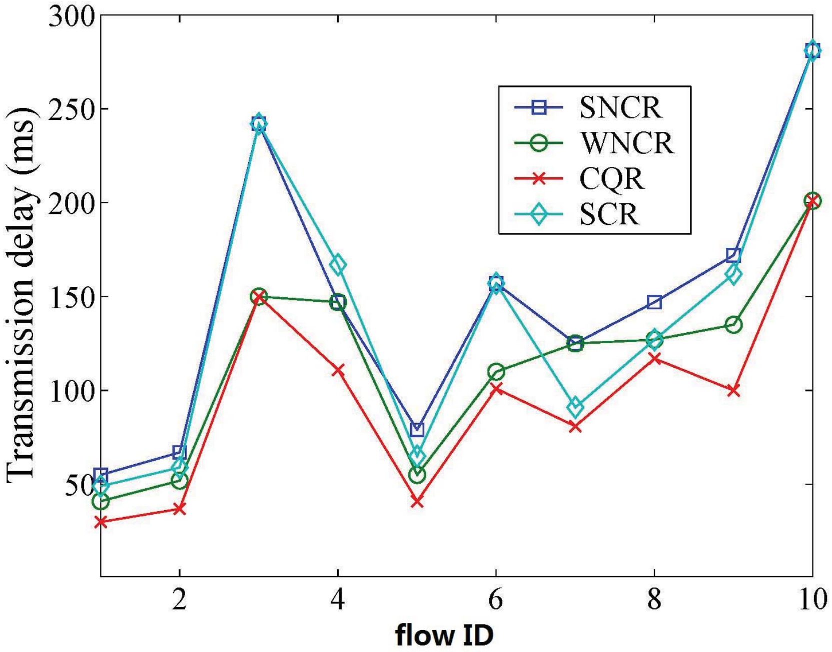


Fig. 3.17: Comparison of transmission delay in outdoor network

other cases compared with the SNCR and WNCR protocols. This shows that the metric of link quality is not always a good metric to select helper nodes. We also compare the packet loss ratio using different routing protocols as shown in Fig.3.18. The result shows that CQ-routing protocol at most reduces the packet loss ratio by over 41% compared with the SNCR protocol. The result also show that CQ-routing protocol at most reduces the packet loss ratio by 36% compared with the WNCR protocol, and 47% compared with the SCR protocol. This result shows that the CQ-routing protocol can improve the reliability, in term of reducing the packet loss ratio of multi-hop wireless networks. In summary, the CQ-routing protocol can improve the performance of multi-hop wireless networks.

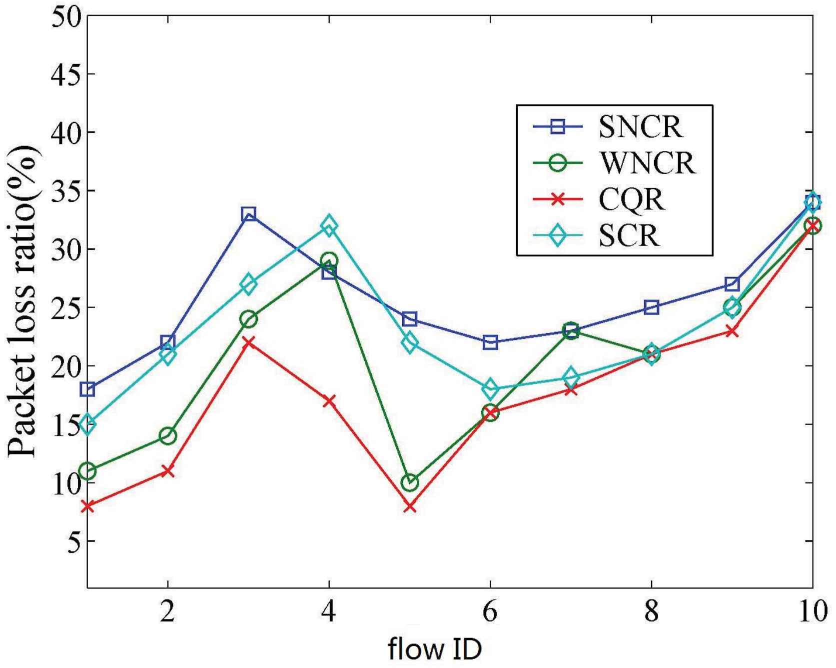


Fig. 3.18: Comparison of packet loss ratio in outdoor network

Evaluation of Admitted Flows

In this subsection, we evaluate the performance of CQ-routing and WNC protocols by number of admitted flows. This metric represents the number of flows whose bandwidth demand is satisfied. In our experiment, admission control is combined with CQ-routing protocol. If available bandwidth routing path selected by CQ-routing protocol is larger than the new connection's bandwidth demand, the new connection is admitted. Otherwise, it will be blocked. We randomly select 40 traffic source and destination node pairs, and set up traffic flows among them. These traffic flows come one by one. We use different routing protocols to select routing paths for these flows. We test available bandwidth for these flows. We stop testing these flows Once one traffic flow's bandwidth demand is not satisfied. We count the number of flows whose bandwidth demand is satisfied when using different routing protocols. We assume

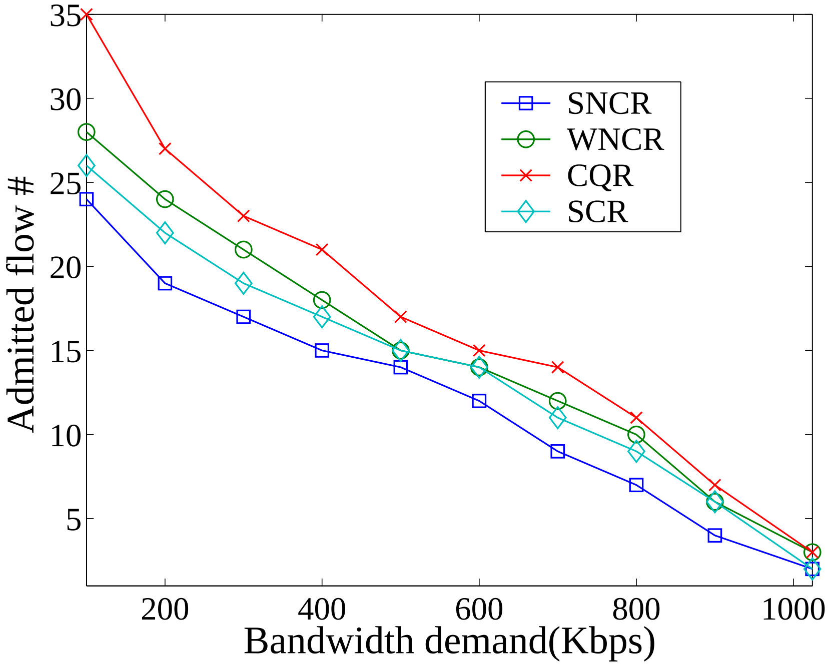


Fig. 3.19: Comparison of number of admitted flows in outdoor network

that each flow has the same bandwidth demand.

To test the impact of bandwidth demand, we test the number of admitted flows with different bandwidth demands, which varies from 100Kbps to 1Mbps. As we can see from Fig.3.19, the CQ-routing protocol can at most admit 11 more traffic flows compared with the SNCR protocol, and 9 more traffic flows compared with the WNCR protocol. This is because the CQ-routing protocol considers interference and cooperative communication issue, so it can find cooperative routing path with more available bandwidth. This improves the opportunity that one flow's bandwidth demand is satisfied. Moreover, the number of admitted flows decreases as the increase of bandwidth demand. This is reasonable because the bandwidth resource is limited, the increase of bandwidth requirement leads to decrease of admitted flows.

Evaluation of Routing Discovery Time

To evaluate the overhead of CQ-routing protocol, we compare the routing discovery time of CQ-routing protocol with that of three other routing protocols. Routing discovery time denotes the duration when the source node starts the connection request until the source node starts to transmit the first packet. We set up multiple flows with different hops. We test each connection with different routing protocols 100 times, and calculate the average value as the final result.

As we can see from Fig.3.20, CQ-routing protocol increases the routing discovery time by about 40% compared with shortest-path routing protocol. The is because that CQ-routing protocol increases the size of control messages, like RREQ, RREP message, and brings new message like HELP message. It also increases the computation time to find helpers. Although CQ-routing protocol increases the routing discovery time, we argue that the increased routing discovery time is still tolerable since it only needs to find cooperative routing path when it starts the traffic flow.

3.5.4 Simulation Results

In this subsection, we evaluate the performance of both centralized and distributed CQ-routing algorithms in different networks through simulations. We set up a multi-hop wireless network with 100 nodes, which are randomly distributed in the network. The related configuration parameters are shown in TABLE 3.3. We assume each link can achieve the Shannon capacity in subsection 3.2.1. We compare the available bandwidth and admitted flows of different cooperative routing protocols. We also analyze the impact of number of hops and node density to the performance.

We compare available bandwidth of different cooperative routing protocols. We

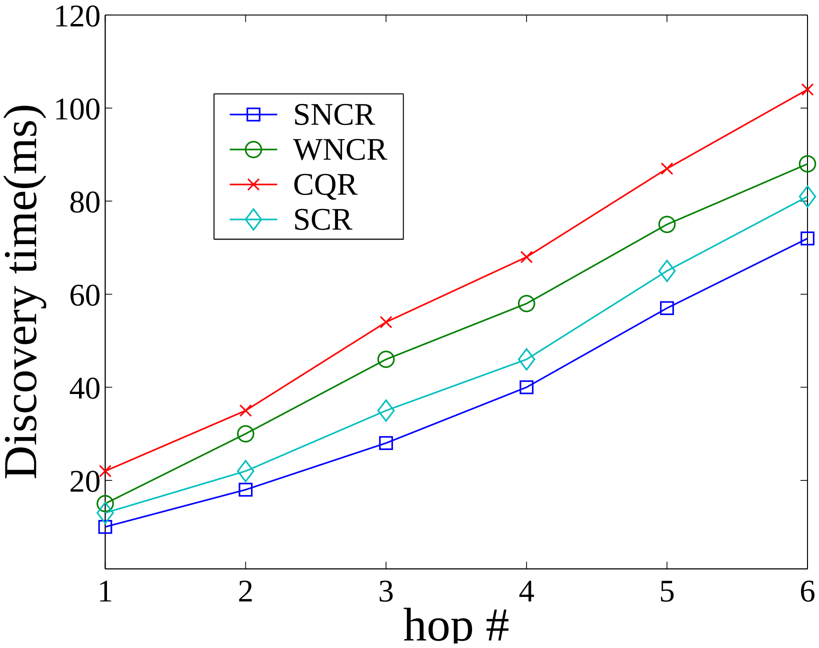


Fig. 3.20: Comparison of routing discovery time in outdoor network

randomly select 10 source and destination node pairs. We test available bandwidth of cooperative routing paths selected by different routing protocols shown in Fig .3.21. From the results, we find that the Cen-CQ algorithm achieves maximum available bandwidth. It can improve the available bandwidth by 186.35%, 90.91% compared the SNCR and the WNCR algorithms. This is because the Cen-CQ algorithm selects the routing path with high available bandwidth, and adopt the cooperative communication technology to further improve poor links' transmission capacity. Compared with the WNCR algorithms, the Dist-CQ algorithm improves the available bandwidth by 27.89% . This is because the Dist-CQ algorithm adopts the cooperative communication technology to further improve poor links' transmission capacity. The Cen-CQ algorithm achieves more available bandwidth than the Dist-CQ algorithm because it has global information. Moreover, although the SCR algorithm also adopts

Table 3.3: Parameters for Simulations

Parameters	Values
Noise level (N_0)	-70dBm
Transmission power (P_0)	16dBm
Transmission distance (d_0)	60m
Path loss exponent (α)	4
Channel bandwidth (W)	2MHZ
Bandwidth demand (RBW)	500Kbps

the cooperative communication technology, it sometimes achieves even worse performance than the SNCR algorithm. This is because it does not consider transmission capacity and available time when selecting helper nodes. This shows the importance of appropriately selecting helper nodes.

We test the number of admitted flows with different bandwidth demand shown in Fig .3.22. We randomly select 30 source and destination node pairs, and select cooperative routing paths for these flows using different routing protocols. If one flow's bandwidth demand is not satisfied, we stop the testing. The result in Fig .3.22 shows that the network can admit maximum number of flows when using the Cen-CQ algorithm. This is because the Cen-CQ algorithm can find cooperative routing path with large available bandwidth. This improves the opportunity that one traffic flow is adopted. Moreover, compared the WNCR algorithm, the Dist-CQ algorithm can at most admit 4 more flows. It improves the number of admitted flows by 30.76% compared with the WNCR algorithm. Moreover, the number of admitted flows decreases as the increase of bandwidth demand. This is because single flow with larger bandwidth demand occupies more bandwidth resource. We test the number of admitted flows with different number of hops. We select multiple sets of source and destination node pairs with different number of hops. The first set of source and destination node pairs have the shortest path with hop number of 2, and the second set of source and destination node pairs have the shortest path with hop number of 3. Each set has 30 source and destination node pairs. All these flows have the bandwidth demand of 1200Kbps. The result in Fig .3.23 shows that the number of admitted flows decreases as the increase of number of hops. This is because when the flow with more number of hops consumes more bandwidth of the network, and increase the opportunity of interference. This makes the number of admitted flows decrease as the increase of number of hops.

We analyze the impact of node density to the performance of centralized and distributed CQ-routing algorithm. We set up multiple networks, in which wireless nodes are uniformly distributed. We test available bandwidth and number of admitted flows in these networks with different node density. The node density is measured by the average distance between neighboring nodes. In each network topology, we randomly select 30 source and destination node pairs, and select cooperative routing paths for these flows with different cooperative routing protocols. We calculate the average available bandwidth of these cooperative routing paths shown in Fig .3.24. The result shows that the available bandwidth decreases as the increase of average distance between neighboring nodes. We explain this phenomenon as follows. Because of path loss effect, the SINR decreases as the distance between neighboring nodes increases. This leads to the decrease of links' transmission capacity and available bandwidth.

Moreover, the performance gain achieved by centralized and distributed CQ-routing algorithms increases as the increase of average distance between neighboring nodes. For example, when the distance is 30m, the centralized and distributed CQ-routing algorithms achieve almost the same available bandwidth as that of SNCR algorithm. When the distance is 55m, the centralized and distributed CQ-routing algorithms improve the available bandwidth by 42.72% compared with the SNCR algorithm. We explain the phenomenon as follows. When the distance between neighboring nodes is very close, the link quality between neighboring nodes is good so the cooperative communication can not provide any performance gain compared with direct transmission. When the distance between neighboring nodes becomes large, the link quality between neighboring nodes becomes worse so the cooperative communication can provide performance gain compared with direct transmission.

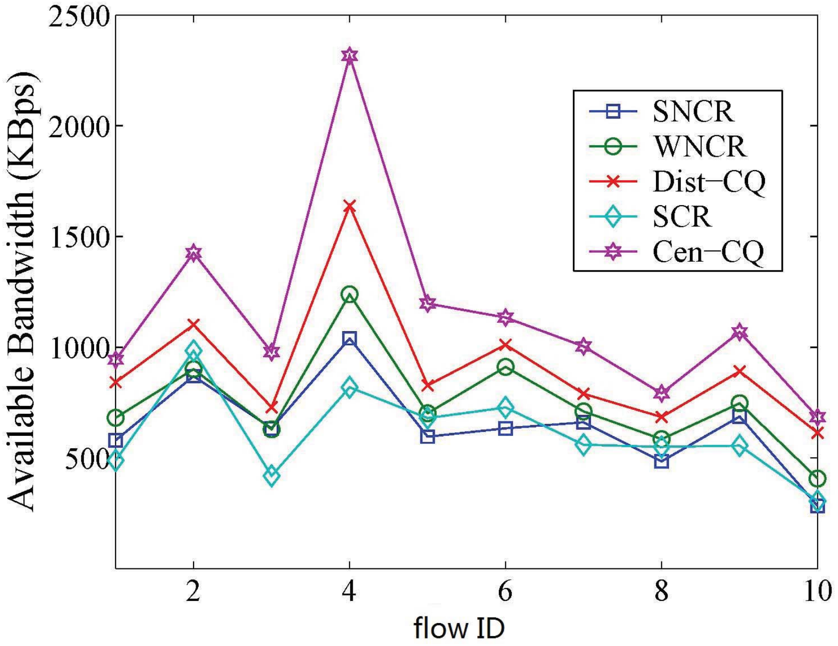


Fig. 3.21: Comparison of available bandwidth

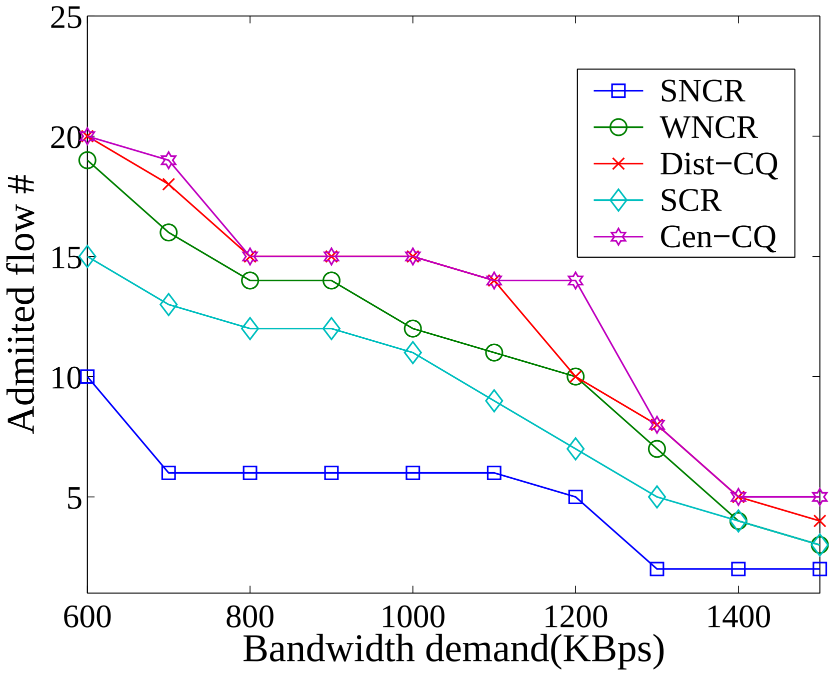


Fig. 3.22: Comparison of number of admitted flows VS bandwidth demand

We also test the number of admitted flows in multiple networks with different node density shown in Fig .3.25. We select 30 traffic flows, which all have the shortest path of the same length, 3 hops, bandwidth demand=1024Kbps. The results show that the number of admitted flows increase as the average distance between neighboring node. This is contrary to previous result in Fig .3.24 since the available bandwidth decreases as the average distance between neighboring node. We explain the phenomenon as follows. When the distance between neighboring nodes is close, the interference effect becomes serious among links. When multiple flows come, the interference effect plays a dominant role when deciding whether or not these flows should be admitted. The result also shows that centralized and distributed CQ-routing algorithms achieve more performance gain when the average distance between neighboring nodes increases. Therefore, the centralized and distributed CQ-routing algorithms are more applicable

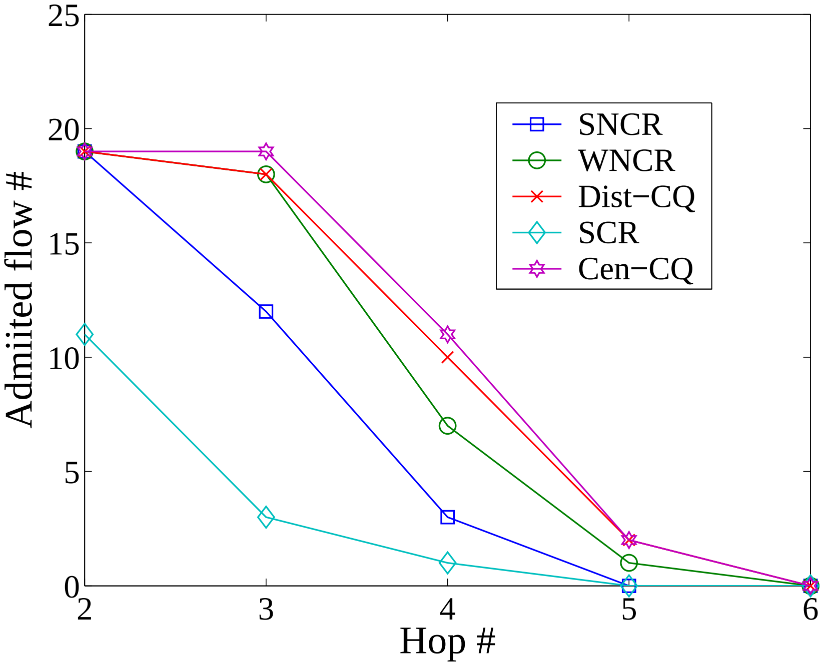


Fig. 3.23: Comparison of number of admitted flows VS number of hops

in the multi-hop wireless network with low node density.

3.6 Conclusion

In this paper, we design a cooperative QoS routing protocol for a multi-hop wireless networks. We propose both centralized and distributed algorithms to solve the Coop-routing problem. Based on the distributed Coop-routing algorithm, we design a cross-layer cooperative QoS routing protocol. We implement the CQ-routing protocol in our HAWK mesh testbed. We evaluate the performance of the CQ-routing protocol by experiments and simultations. The results show that CQ-routing protocol can significantly improve the network performance. Although the routing discovery time becomes longer, we argue that it is tolerable since one flow only needs to find a

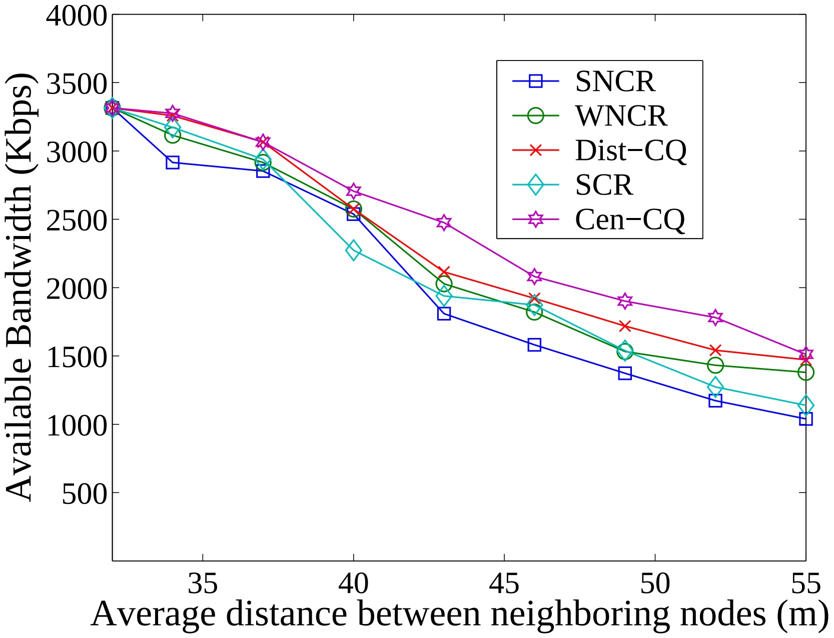


Fig. 3.24: Comparison of available bandwidth VS node density

cooperative routing path once.

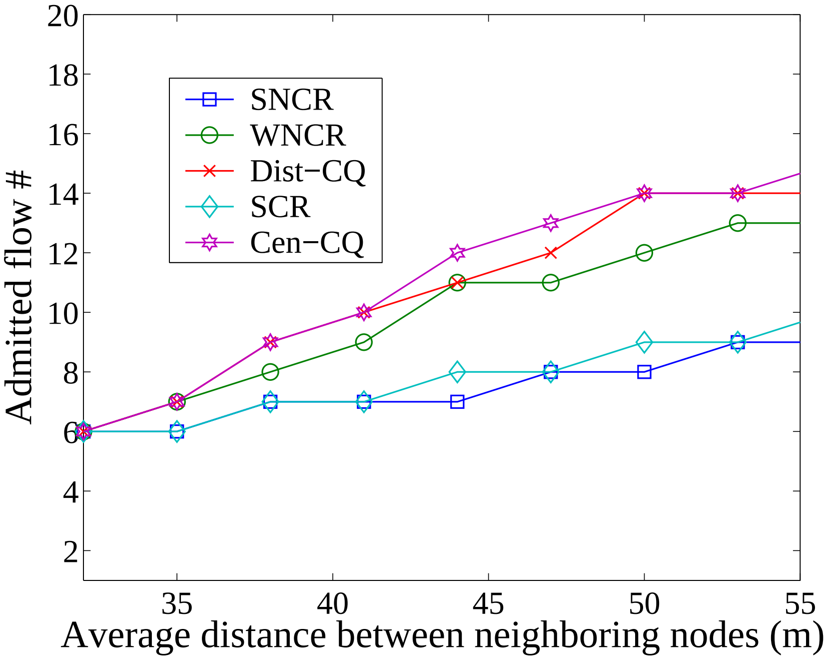


Fig. 3.25: Comparison of number of admitted flows VS node density

Chapter 4

Scheduling of Multi-link Spectrum Handoff in Multi-radio Multi-hop Cognitive Networks

4.1 Introduction

Cognitive radio is an emerging technology to solve the frequency scarcity problem through dynamic spectrum access [Akyildiz et al., 2006], [Kyasanur and Vaidya, 2005]. In cognitive networks, there are two kinds of users, namely primary users and secondary users. Primary users have the license to access the spectrum, while secondary users do not have that license. By using the cognitive radio technology, secondary users can use the licensed spectrum of primary users under the constraint of not interfering primary users' communication. When primary users reclaim their rights or spectrum quality becomes worse, communication links between secondary users need to find a new available spectrum and switch to it, called spectrum handoff

[D.Cohen, 2005]. Hereafter, we name the links which need to perform spectrum handoff by switching links, and those links which do not need by normal links. When links perform spectrum handoff, their communication is interrupted. There are two kinds of spectrum handoff, reactive and proactive spectrum handoff [Yu and Richard, 2011]. In reactive spectrum handoff, switching links perform spectrum handoff after detecting link failures. This method requires immediate spectrum handoff without any preparation time, resulting in significant performance degradation. In proactive spectrum handoff, secondary users can predict future events on the current link, and perform spectrum handoff before the current spectrum becomes unaccessible [Yu and Richard, 2011]. The proactive spectrum handoff is usually triggered by two events, including appearance of primary user and degradation of spectrum quality. The former event can be predicted by priori agreements about spectrum occupation time or accurate models in [Yang et al., 2007]. The latter event is easy to detect, and it does not require the secondary users to perform spectrum handoff immediately. In this paper, we only consider proactive spectrum handoff, and assume that secondary users have enough time to prepare for spectrum handoff.

Existing works only studied the spectrum handoff problem for a single-link in a single-hop cognitive network, referred to as the SH-SLSH problem. It is concerned with minimizing single link's spectrum handoff delay. The spectrum handoff delay is defined as the time period when the communication of one switching link is interrupted by spectrum handoff. It depends on: (1) spectrum discovery time: the time for secondary users to find new spectrums by performing spectrum sensing; (2) switching time: the time for the RF front-end hardware to reconfigure its operating spectrum;

(3) link establishment time: the time for secondary users to find neighbors and configure transmission parameters. The latter two kinds of time is inevitable, so existing works focus on designing spectrum sensing algorithms to reduce the spectrum discovery time. This paper studies a more challenging problem in which multiple links perform spectrum handoff in multi-radio multi-hop cognitive networks, referred to as the SH-MLMH problem. In the multi-radio multi-hop cognitive network, the communication of switching links can be kept uninterrupted during spectrum handoff by maintaining the network connectivity and rerouting. This eliminates the spectrum handoff delay. Assuming an existing routing protocol is adopted to perform rerouting, the SH-MLMH problem first targets at maintaining network connectivity by scheduling links to perform spectrum handoff. However, even if the network connectivity is maintained, the network still suffers from minor throughput degradation during spectrum handoff because spectrum handoff leads to the sub-optimal transmission. Thus, while maintaining the network connectivity, we also should minimize the Total Handoff Completion Time (THCT) such that the network quickly recovers its optimal transmission. The THCT is defined as the total time for all switching links to finish spectrum handoff. Notice that the THCT is different from the spectrum handoff delay because the communication is not interrupted but just sub-optimal during it. The THCT depends on: (1) spectrum handoff time for single link: the time that decides single link's spectrum handoff delay; (2) rerouting time: the time for routing protocols of switching links to find alternative routing paths and converge; (3) the order for multiple links to perform spectrum handoff: some links can concurrently perform spectrum handoff to reduce the THCT. We assume that an existing spectrum sensing algorithm is adopted to find new spectrum, and the spectrum handoff and

rerouting time for single link is constant. Thus, THCT only depends on the order for multiple links to perform spectrum handoff, so this paper minimizes the THCT by scheduling multiple links to perform spectrum handoff. Considering these two new issues, we formulate the SH-MLMH problem as a new optimization. It targets at scheduling links to perform spectrum handoff such that the network connectivity is always maintained and the THCT is minimized.

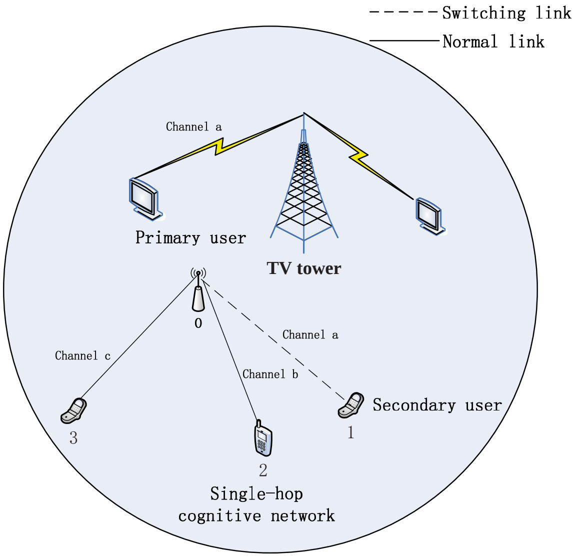


Fig. 4.1: Example of SH-SLSH problem.

We explain the difference between the SH-SLSH problem and the SH-MLMH problem by examples in Fig. 4.1 and Fig. 4.2, which respectively illustrate a single-hop cognitive network and a multi-hop one. Both scenarios include one primary network (TV towers and TV sets) and one secondary network (Base Station and mobile devices). In the single-hop scenario, when the TV tower is idle, node 1 occupies its spectrum (channel a) to transmit packets. When the TV tower becomes active, link l_{01} needs to perform spectrum handoff, and its communication is interrupted. In the multi-hop scenario, each mobile device is equipped with two radios. The

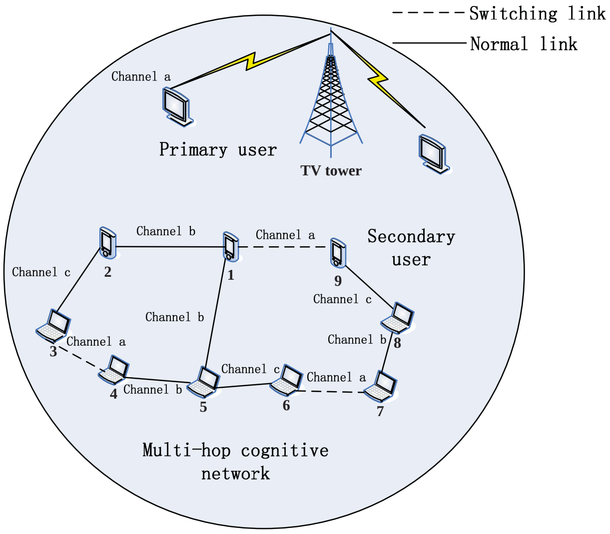


Fig. 4.2: Example of SH-MLMH problem

licensed spectrum of the TV tower (channel a) is occupied by multiple mobile devices. Moreover, the transmission power of the TV tower is much larger than that of mobile devices. When the TV tower becomes active, it will influence multiple mobile devices in a large area, and cause three links l_{19}, l_{34}, l_{67} to perform spectrum handoff. If they perform spectrum handoff simultaneously, the secondary network will become partitioned, and the communication of switching links is interrupted. If these three links perform spectrum handoff sequentially, the network is kept connected, and the routing protocol can be triggered to find alternative routing paths to transmit packets. For example, if only link l_{19} performs spectrum handoff, it can transmit packets through another routing path, $P : 1- > 5- > 6- > 7- > 8- > 9$. However, the transmission from node 1 to 9 through path P is not optimal compared with direct transmission, so we should speed up the process that the network recovers its optimal configuration. This requires us to minimize the THCT by scheduling links to perform spectrum handoff. For example, if links l_{19} and l_{34} concurrently perform spectrum handoff at first, and link l_{67} performs spectrum handoff later, the network is still kept connected during spectrum handoff and the THCT is reduced.

We make the following main contributions in this paper. (1) We formally formulate the SH-MLMH problem, and prove that it is NP-hard. (2) We propose a centralized algorithm to solve this problem. Hereafter, this algorithm is referred to as the centralized Spectrum Handoff Scheduling (SHS) algorithm. We prove that the centralized SHS algorithm can achieve a logarithmic approximation ratio. (3) Because the centralized SHS algorithm requires global information, we propose a distributed algorithm requiring only local information to overcome this disadvantage. Hereafter, this algorithm is referred to as the distributed Spectrum Handoff Scheduling (SHS) algorithm. We prove that the distributed SHS algorithm generates valid solution. Through a cross-layer message mechanism, the scheduling algorithms are also responsible to coordinate routing protocols to perform rerouting such that the rerouting time is reduced. We also propose a rerouting algorithm which cooperates with the SH-MLMH algorithm to find rerouting paths. The simulation results show that both SHS algorithms not only improve system throughput during spectrum handoff but also reduce THCT compared with a simple solution in which all links perform spectrum handoff simultaneously. We also find that the performance of our SHS algorithms is very close to that of the optimal solution, and they can achieve the same performance as the optimal solution when each node is equipped with enough radios.

The rest of the paper is organized as follows. Section 2 introduces the related work. In Section 3, we describe the system model, formulate the SH-MLMH problem, and prove that the SH-MLMH problem is an NP-hard problem. Section 4 presents the detailed design of SHS algorithms, and analyzes the performance of the SHS algorithms. In section 5, we evaluate the performance of SHS through simulations. Finally, we conclude this work and present the future work in Section 6.

4.2 Problem Formulation

4.2.1 System Model

We model the multi-hop cognitive network as an undirected Graph $G(V, E)$, where V refers to a set of cognitive nodes, and E refers to a set of links between cognitive nodes. We define the terminologies used in this paper in Table 4.1. In this paper, we make following assumptions.

We assume that each cognitive node is equipped with k radios, $k > 1$. We assume that an existing channel assignment algorithm is adopted to assign channels to these radios, and each radio is only assigned one channel. One link is established between one node pair if these two nodes are within each other's communication range, and they have radios working on the same channel. We assume that each switching link performs proactive spectrum handoff [Pei et al., 2007, Yu and Richard, 2011], in which secondary users can have enough time to prepare for spectrum handoff in advance. We assume out-of-band spectrum sensing is adopted during spectrum handoff, and secondary users only start to find new spectrums after spectrum handoff begins. We assume that an existing routing protocol is adopted to perform rerouting, and it can find routing paths between any node pair if the network is connected. To facilitate design of solutions, we further assume that the spectrum handoff and rerouting time for single link is constant. If multiple links perform spectrum handoff sequentially

Table 4.1: Terminologies

Short Term	Terminology	Description
HLS	Handoff Link Set	The set of links, which need to do spectrum handoff
T	Reserved Spanning Tree	The selected spanning tree of Graph $G(V, E)$, whose links are not removed
S_i	The *ith* switch link set	The set of switching links that are scheduled to perform spectrum handoff in the *ith* round
LAT	Left Available Time	LAT of spectrum s for link l is defined as the time link l can work on spectrum s before it needs do spectrum handoff.
DS	Dominating Set	Given a graph $G(V, E)$, Dominating Set D is defined as a subset D of V such that every vertex not in D is joined to at least one member of D by some link.
CDS	Connected Dominating Set	A connected dominating set of a graph $G(V, E)$ is a set of vertices $D \in V$ such that D is a dominating set of G, and the sub-graph induced by D is connected.
AST	Average System Throughput	AST is defined as the average total data rates achieved by all the traffic flows during spectrum handoff.
$TPLR$	Total Packet Loss Ratio	$TPLR$ is defined as the ratio of total lost packets to total sent packets during total spectrum handoff time.
$THCT$	Total Handoff Completion Time	$THCT$ is defined as the total time for all links to finish spectrum handoff.

in multiple rounds, we assume THCT depends on the number of rounds and the spectrum handoff and rerouting time for single link.

$$T_M = \bar{T} * N \tag{4.1}$$

Here, $\bar{T}$ is the spectrum handoff and rerouting time for single link and N is the number of rounds. The THCT depends only on the number of rounds. The switching link can recover its communication after they perform spectrum handoff.

4.2.2 Problem Formulation

We formally formulate SH-MLMH problem as the following optimization problem.

Definition1. (SH-MLMH problem) Given a graph $G(V, E)$, and a link subset $HLS \subseteq E$, our problem is to find a sequence of link subsets of HLS, $S = \{S_1, S_2, ... | S_i \subseteq HLS\}$, which satisfy following constraints.

Objective: $\min(|S|)$

Subject to.

1. $\forall S_i \in P$, sub-graph $G_i(V, E - S_i)$ is a connected graph;
2. $\cup S_i = HLS$;
3. $S_i \cap S_j = \emptyset$, if $S_i \in S, S_j \in S, S_i \neq S_j$;

Here, the objective of minimizing total handoff time is transformed to minimize the number of subsets, which denote the number of rounds that all switching links need to perform spectrum handoff. The first constraint denotes the network connectivity constraint. The second constraint denotes that each link in HLS must finish its spectrum handoff. The third constraint denotes that single link just needs to perform spectrum handoff once.

4.2.3 Analysis of Computation Complexity

In this subsection, we prove that SH-MLMH problem is an NP-hard problem. We first prove that SH-MLMH problem can be modeled as a disjoint set cover problem. Then we prove that disjoint set cover problem is an NP-hard problem by reducing it to a well-known NP-hard problem, set cover problem [M.R.Garey and D.S.Johnson, 2005]. We define set cover problem and disjoint set cover problem as follows.

Definition2. (Set cover problem) Given a finite set, $S = \{s_1, s_{2,}...\}$, and a collection C of subsets of S,$C = \{C_1, C_{2,}...|C_i \subseteq S\}$, the problem is to find a minimum subset $C^{'}$ of C such tat every element of S belongs to *at least one* member of $C^{'}$.

Definition3. (Disjoint set cover problem) Given a finite set S, $S = \{s_1, s_{2,}...\}$, and a collection C of subsets, $C = \{C_1, C_{2,}...|C_i \subseteq S\}$, the problem is to find a minimum subset $C^{'}_{disj}$ of C such that every element of S belongs to *one* member of $C^{'}_{disj}$,

.

Lemma1. SH-MLMH problem can be modeled as a disjoint set cover problem.

Proof. We prove this by modeling SH-MLMH problem as a disjoin set cover problem. We first list all the link subsets satisfying the connectivity constraint. To simplify the presentation, we denote the link subset satisfying the connectivity constraint by feasible link subset. Given a graph $G(V, E)$, we can calculate all the spanning trees of G within polynomial time [Novak et al., 2005], $T = \{T_1, T_2, ..\}$. We can calculate the maximal feasible link subsets satisfying the connectivity constraint, $S_i = (E - T_i) \cap HLS$. Here, maximal feasible link subset denotes the feasible link subset which is not a subset of any other feasible link subset. In other words, if we add any other link to a maximal feasible link subset, it can not satisfy the connectivity constraint. Obviously, subset s of each maximal feasible link subset is also a feasible link subset. Then we can list all the feasible link subsets as the union of each maximal feasible link subsets, $\cup_i Power(S_i)$. We denote the collection of all feasible link subset by feasible link subset space. Therefore, SH-MLMH is corresponding to a disjoint set cover problem by setting the finite set S of disjoint set cover problem equal to HLS, and the collection of subsets C equal to $\cup_i Power(S_i)$. □

Theorem1. SH-MLMH is an NP-hard problem.

Proof. Because the SH-MLMH problem is modeled as a disjoint set cover problem, we prove that SH-MLMH is an NP-hard problem by reducing the set cover problem to the disjoint set cover problem. Given an instance M of set cover problem, a finite set S,$S = \{s_1, s_{2,}...\}$, and a collection C of subsets of S, $C = \{C_1, C_{2,}...|C_i \subseteq S\}$, we construct an instance $M^{'}$ for a disjoint set cover problem based on the same finite set S and a collection of subsets of S, C_{disj}. We construct C_{disj}, as follows. First, C_{disj} includes all the subsets in C, $C \subseteq C_{disj}$. Then for any two subsets in C, $C_i, C_{j,}$, if $C_i \cap C_j \neq \emptyset$, we add a new subset $C_i^{'} = C_i - C_i \cap C_j$ to $C^{'}$. $C^{'}$ can be calculated as follows $C^{'} = C \cup (\cup_i C_i^{'})$. Now we prove that there exists a set cover $C^{'} = \{C_1, C_2, ..., C_k\}$ with K elements for instance M if and only if there exists a set cover $C_{disj}^{'} = \{C_1^{'}, C_2^{'}, ..., C_k^{'}\}$ with K elements for instance $M^{'}$. Given an optimal solution for set cover problem, which is a set cover $C^{'} = \{C_1, C_2, ..., C_k\}$ with K elements for instance M, we calculate a set cover $C_{disj}^{'} = \{C_1^{'}, C_2^{'}, ..., C_k^{'}\}$

with K elements for instance $M^{'}$ as follows. If for any $C_i \in C^{'}$, if for any other $C_j \in C^{'}$, $C_i \cap C_j = \emptyset$, we set $C^{'}_{disj} = C^{'}_{disj} + C_i$. Otherwise, if $\exists C_j \in C^{'}$, such that $C_i \cap C_j \neq \emptyset$, due to above construction of C_{disj}, there must exist a sub-set $C^{'}_i \in C_{disj}$, $C^{'}_i = C_i - C_i \cap C_j$, we set $C^{'}_{disj} = C^{'}_i + C^{'}_{disj}$. It's to verify that $C^{'}_{disj}$ covers all elements of S, and the cardinality of $C^{'}_{disj}$ is K. Similarly, given a disjoint set cover $C^{'}_{disj} = \{C^{'}_1, C^{'}_2, ..., C^{'}_k\}$ with K elements for instance $M\prime$, we can also calculate a set cover with K elements for instance M. Moreover, all the transformation is within polynomial time. Because the set cover problem is an NP-hard problem, the disjoint set cover problem is an NP-hard problem. In conclusion, we prove that SH-MLMH problem is also an NP-hard problem. □

4.3 The *SHS* Algorithms

4.3.1 Overview

Before we present detailed SHS algorithms, we explain how they cooperate with routing protocols to perform spectrum handoff. Given a cognitive network $G(V, E)$ and a Handoff Link Set (HLS), following main steps are performed to coordinate multi-link spectrum handoff in Fig. 4.3.

- ***SHS***: SHS algorithms decide the order for multiple links to perform spectrum handoff. The HLS will be divided into a sequence of link subsets, $S = \{S_1, S_2, ... | S_i \subseteq HLS\}$. Here, S_i is the subset of links which perform spectrum handoff in the ith round, called the ith Switch Link Set.
- **Rerouting:** Given the scheduling results of SHS, links in S_i perform rerouting to find alternative routing paths under the constraint that the alternative paths

do not include switching link in S_i. The SHS algorithms are responsible to trigger and inform the routing protocol links in S_i. To avoid its adverse impact on communications, the routing protocol is triggered to perform rerouting before spectrum handoff happens.

- **Spectrum Handoff:** After switching the traffic flow to alternative routing path, links in S_i perform spectrum handoff by finding and switching operating spectrums to new spectrums. The HLS is updated, and these three stages are run iteratively until all links finish spectrum handoff.

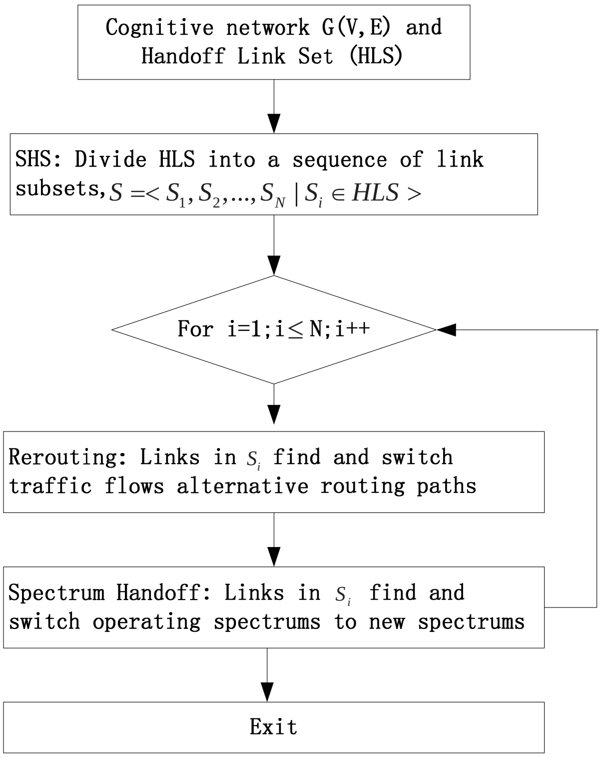


Fig. 4.3: Flow chart of multi-link spectrum handoff coordination

4.3.2 Centralized *SHS* Algorithm

In this subsection, we propose a centralized Spectrum Handoff Scheduling algorithm (SHS) to solve SHS sub-problem and analyze its performance. The centralized SHS algorithm targets at minimizing THCT while keeping the network connectivity. The input parameter to the SHS algorithm is the network $G(V, E)$ and Handoff Link Set HLS. The output result of the scheduling algorithm is a sequence of switching link set $S = \{S_1,S_2, ...|S_i \subseteq HLS\}$.

The basic idea of centralized SHS algorithm is described as follows. We divide links in HLS into multiple disjoint subsets. Links in the same subset perform spectrum handoff simultaneously, and links in different subsets perform spectrum handoff in different rounds. To minimize the THCT is equal to minimize the number of the subsets, $minimize|S|$. Intuitively, we can minimize $|S|$ by maximizing each $|S_i|$ under the constraints of network connectivity. In each round, the problem is transformed to maximize each Switch Link Set. Considering the connectivity constraint, for each S_i, sub-graph $G_i(V, E - S_i)$ needs to be connected. As we know that a spanning tree is the minimum sub-graph to keep the connectivity of a graph, so a spanning tree needs to be built up as Reserved Spanning Tree T_i. Here, T_i denotes a spanning tree of $G(V, E)$, whose links do not perform spectrum handoff in the ith round. The problem of maximizing S_i is transformed to following problem:

$$\max(|S_i|) = \max_{T_i} |(E - T_i) \cap HLS| = |HLS| - \min_{T_i}(|T_i \cap HLS|) \tag{4.2}$$

Therefore, to maximize S_i is transformed to find a spanning tree T_i with minimum number of links in HLS. We find such a spanning tree T_i by assigning different weights to links in HLS and links not in HLS. Here, we assign 1 as weight of links in HLS, and 0 as weight of links not in HLS. Then, we calculate the minimum weighted spanning tree T as the ith Reserved Spanning Tree T_i. After we calculate the ith Reserved Spanning Tree T_i, we calculate the ith Switch Link Set S_i based on T_i. Then links in S_i perform spectrum handoff, and we update HLS. We run above procedures iteratively until all links in HLS finish the spectrum handoff. The detail of the centralized SHS algorithm is described in Algorithm 4.

Algorithm 4: CENTRALIZED SHS ALGORITHM

```
Input: Handoff link set $HLS$, Graph $G(V, E)$
Output: Sequence of switch link set $S$
while $HLS$ is not null do
    Assign weight to each link, $\forall l_i \in HLS, weight_i = 1; \forall l_i \notin HLS, weight_i = 0$;
    Calculate the minimum spanning tree $T_i$ of G ;
    Calculate the Switch Link Set,$S_i = (G - T_i) \cap HLS$;
    Wait for the spectrum handoff of links in $S_i$ to be finished ;
    Update $HLS$,$HLS = HLS - S_i$ ;
    $i++$;
```

The centralized SHS algorithm is run in a central server. We design a message mechanism in Fig 4.4 to show how the central scheduling server coordinates cognitive nodes to perform rerouting before spectrum handoff happens. Any cognitive node which needs to perform spectrum handoff sends a $REQUEST$ message to the central server to ask for the token. The central server will update HLS, and calculate S_i, and multiple tokens are granted to the corresponding nodes of the links in S_i. The token includes links in S_i, indicating which links are scheduled to perform spectrum handoff in this round. If a cognitive node receives a token, it starts the rerouting process to find a new routing path without the links in S_i. The traffic will be switched to the new path. After that, the cognitive node sends a *Ready_to_Switch* message to the scheduling server to indicate that the traffic switch has been finished. If the scheduling server collects all the $Ready\_to\_Switch$ messages from the corresponding nodes in S_i, it sends ACK_Switch message to links in S_i, and the links receiving ACK_Switch message perform spectrum handoff. After the spectrum handoff is finished, the cognitive nodes send $Finish\_Switch$ message to the scheduling server. If the scheduling server collects all the $Finish\_Switch$ messages, it updates the HLS, and starts scheduling of another round until all links in HLS have finished their spectrum handoff. In this way, the spectrum handoff of multiple links is synchronized so that all links do spectrum handoff after rerouting.

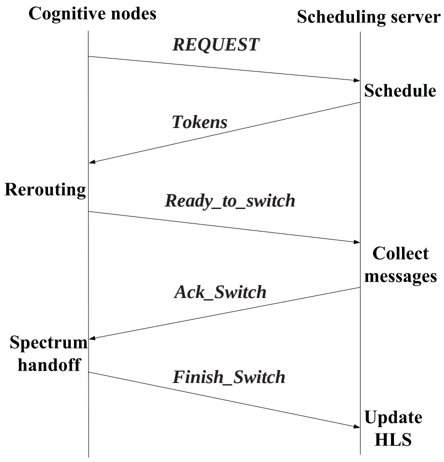


Fig. 4.4: Message mechanism for coordinating rerouting

We analyze the time complexity and efficiency of the centralized SHS algorithm as follows. For time complexity, we prove that the time complexity of the centralized SHS algorithm is polynomial. For efficiency, we have following main conclusions. Given a k-link-connected graph $G(V, E)$, if $k = 1$, SH-MLMH problem is not solvable. If $k = 2$ or 3, the centralized SHS algorithm can achieve a logarithmic approximation ratio of $ln|HLS|$. If $k \geq 4$, the centralized SHS algorithm can achieve an optimal solution. We describe the detailed proof as follows.

Theorem2. Given a cognitive network $G(V, E)$ and HLS, the time complexity of centralized SHS algorithm is $O(min(|V|, |HLS|) * |E| * \alpha(|V|, |E|))$. Here, α is the classical functional inverse of the Ackermann function [Chazelle, 2000].

Proof. As we can see from Algorithm 4, in each iteration, most time is spent in calculating the reserved Minimum Spanning Tree (MST), which takes more time than other step. The fastest minimum spanning tree algorithm is developed by Bernard Chazelle in [Chazelle, 2000]. Its time complexity is

$$O(|E| * \alpha(|V|, |E|)) \tag{4.3}$$

We denote the total number of iterations for while loop in centralized SHS algorithm by L. At the worst case, only one switching link performs spectrum handoff in each iteration. We can find that :

$$L \leq |HLS| \tag{4.4}$$

As we can see from line 6 of the centralized SHS algorithm, in the first iteration, all switching links except those in the MST perform spectrum handoff. We denote the number of links in MST by $|MST|$. We can find that after the first iteration, at most $|MST|$ switching links still do not perform spectrum handoff. At the worst case, after the first iteration, only one switching link performs spectrum handoff in each iteration. In this way, the maximum number of iterations is $|MST| + 1$. We know that the $|MST| = |V| - 1$. We can find that

$$L \leq |V| \tag{4.5}$$

Combining the results in (4.3), (4.4), (4.5), the total time complexity of the centralized SHS algorithm is $O(min(|V|, |HLS|) * |E| * \alpha(|V|, |E|))$. □

Lemma2. Given a $k - link - connected$ graph $G(V, E)$ and a link subset HLS, if $k = 1$, SH-MLMH problem is not solvable; if $k > 1$, SH-MLMH is solvable.

Proof. Given a $k - link - connected$ graph $G(V, E)$, if $k = 1$, G is partitioned by removal any link from G, so SH-MLMH problem is not solvable. If $k > 1$, we can always solve this problem. At worst case, only one link performs spectrum handoff in one round. Because $k > 1$, the removal of any one link does not render G partitioned. Therefore, SHS problem is always solvable when $k > 1$. □

Lemma3. In each round, the centralized SHS algorithm selects the link subset with maximum left switching links as the switch link set S_i.

Proof. We denote the set of switching links which still do not perform spectrum handoff in the beginning of the *ith* round by HLS_i. As we can see from equation (4), the maximal feasible link subsets under the connectivity constraint in the *ith* round is:$S_i = E \cap HLS_i - T_i \cap HLS_i$. Because $E \cap HLS_i$ is constant, to maximize S_i is equal to minimize $T_i \cap HLS_i$. The centralized SHS algorithm assigns different weights 1 and 0 to the links in HLS_i and not in HLS_i, and find the minimum spanning tree as T_i . Therefore, T_i is the minimum spanning tree with minimum common set with HLS_i , and $T_i \cap HLS_i$ is minimized. so S_i is maximized, and lemma3 is proven. □

Theorem3. Given a $k - link - connected$ graph $G(V, E)$, $k = 2$ or 3, the approximation ratio of centralized SHS algorithm is $ln|HLS|$.

Proof. We consider an instance for SHS problem with a cognitive network $G(V, E)$ and HLS. As we analyze in proof for lemma 1, the problem is transformed to find a family of disjoint subsets $S = \{S_1, S_{2,...}\}$ to cover HLS such that $|S|$ is minimized while $G - S_i$ is a connected sub-graph. Here, S_i denotes the subset of links which perform spectrum handoff in the ith round. We denote the set of switching links which still do not perform spectrum handoff in the beginning of the ith round by HLS_i. We suppose that the optimal solution has m subsets. Therefore, there must exist a switch link subset with at least $|HLS|/m$ switch links because otherwise the optimal solution contains more than m subsets. According to lemma2, in each round, the centralized SHS algorithm selects the link subset with maximum number of left switching links as the switch link subset S_i. So S_i picked by the centralized SHS algorithm has size at least $|HLS|/m$ because it selects the maximum switch link subsets. Therefore, the number of switching links we still have to cover after the first set is picked is

$$|HLS_2| \leq |HLS| - |HLS|/m \leq (1 - 1/m)|HLS| \tag{4.6}$$

By similar method, we can prove that S_2 found by the centralized SHS algorithm has size of at least $|HLS_2|/m$. We find that

$$|HLS_3| \leq (1 - 1/m)|HLS_2| \leq (1 - 1/m)^2|HLS| \tag{4.7}$$

In general, we have

$$|HLS_{i+1}| \leq (1 - 1/m)^i|HLS| \tag{4.8}$$

We suppose that it takes k rounds to cover all links in HLS. By equation (14), we have $|HLS_{k+1}| \leq (1-1/m)^k|HLS|$, and we need this to be less than 1.

$$(1-1/m)^k|HLS| < 1 \tag{4.9}$$

By permutation, we have

$$(1-1/m)^{m*k/m} < 1/|HLS| \tag{4.10}$$

Because $(1-1/m)^m \approx e^{-1}$, we have

$$e^{-k/m} < 1/|HLS| \tag{4.11}$$

and

$$k/m > \ln|HLS| \tag{4.12}$$

So when $k > m * \ln|HLS|$, all links have finished spectrum handoff, k is upper bounded by $m * \ln|HLS|$. Therefore, $k/m \leq \ln|HLS|$, and Theorem 5 is proven. □

Theorem4. If $G(V, E)$ is $k-link-connected$ graph, $k \geq 4$, the centralized SHS algorithm can achieve an optimal solution.

Proof. In graph theory, we have following conclusions [Bondy et al., 2006]. Given a graph $G(V, E)$, if G is $2k-link-connected$, there must exist k link-disjoint spanning trees in G. If G is $4-link-connected$, there must exist two link-disjoint spanning trees, T_1, T_2. We suppose that Request Link Set HLS is equal to E at the extreme case. We select spanning tree T_1 as reserved spanning tree, and the corresponding switch link subset can be calculated as following, $S_1 = E - T_1$. In the second round, we select T_1 as the $2nd$ switch link subset. Because there exist two link-disjoint spanning trees in graph G, sub-graph $G_2(V, E-T_1)$ is a connected graph. Because if all links in E renders G disconnected, we at least need two round to finish spectrum handoff for all links in HLS. This proves that our algorithm achieves the optimal solution. Similarly, when R is not equal to E, if sub-graph $G(V, E-HLS)$ is not connected, both the optimal solution and our greedy centralized SHS algorithm need to spend two rounds in finishing spectrum handoff for all links. If sub-graph $G(V, E-HLS)$ is connected, both the optimal solution and our greedy centralized SHS algorithm need to spend one round finishing spectrum handoff for all links. Therefore, our centralized SHS algorithm can achieve the optimal solution when $G(V, E)$ is $k-link-connected$ graph, $k \geq 4$. □

Theorem5. Given a $k-link-connected$ graph $G(V, E)$ and a link subset HLS, if $k = 1$, SH-MLMH problem is not solvable; if $k = 2$ or 3, the centralized SHS algorithm can achieve an approximation ratio of $ln|HLS|$; If $k \geq 4$, k the centralized SHS algorithm can achieve an optimal solution.

Proof. Because the spectrum handoff delay is constant, the THCT of all the switching links depends on the SHS algorithm. So the centralized SHS algorithm can achieve the same approximation ratio as the centralized SHS algorithm. Combining results in Theorem 4, 5,6, we can prove this conclusion. □

4.3.3 Distributed *SHS* Algorithm

In this subsection, we design a distributed SHS, to solve the SH-MLMH problem and analyze its performance. Similar to the centralized SHS algorithm, we divide Handoff Link Set HLS into multiple link subsets, which do spectrum handoff in different rounds. In each round, we build up a reserved sub-graph T to maintain the network connectivity. The links in HLS, but not in T perform spectrum handoff in this round. The feature of distributed SHS algorithm is that we construct the reserved spanning tree T in a distributed way. We name the node adjacent to links in HLS by switching node and node not adjacent to links in HLS by normal node. We divide the distributed SHS algorithm into three phases.

In the first phase, we modify the marking process in [Bondy et al., 2006] to find a Connected Dominating Set (CDS) for the cognitive network $G(V, E)$. A connected dominating set for a graph $G(V, E)$ is a subset CDS of V such that every node in V is either in CDS or adjacent to at least one node in CDS, and sub-graph of nodes in CDS is a connected graph. We prefer normal nodes to switching nodes when constructing CDS. So all normal nodes mark themselves as $TRUE$, and switching nodes with two partitioned neighbors are marked $TRUE$. All nodes marked $TRUE$ construct CDS.

In the second phase, we select links to connect nodes in and nodes not in CDS to construct a spanning tree T_i to maintain network connectivity. Initially, T_i includes all normal links. Nodes in CDS exchanges Dominator messages with each other. If node v receives Dominator message by a normal link from node u, node v selects node u as its dominator. Otherwise, node v selects node u which it receives the first Dominator message from as its dominator, and adds link l'_{uv} to T_i. After each node in CDS finds its dominator, node v not in CDS selects a node u in CDS as its dominator, and adds link l_{uv} to T_i . Then we select links to connect nodes not in CDS and nodes in CDS. For each nodes v not in CDS, if there exists another node u in CDS, which is connected to v by a normal link l, node v selects node u as its dominator. Otherwise, we select node u in CDS and a switching link l'_{uv} to connect node u and v.

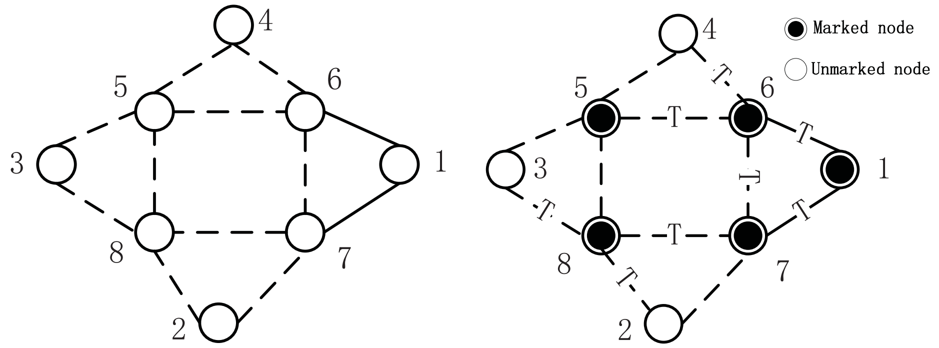
In the third phase, we select links in T but not in HLS as the switching link set S_i. We run these three phases until all links finish spectrum handoff. The detail of the distributed SHS algorithm is described in Algorithm 5.

Algorithm 5: DISTRIBUTED SHS ALGORITHM

```
Input: Network $G(V, E)$, handoff link set $HLS$
Output: Sequence of link subset $S$
while $HLS$ is not null do
    Switching nodes are marked False, and normal nodes are marked TRUE ;
    Normal links are selected as member of $T_i$, $T_i = \{l | l \in E, l \notin HLS\}$;
    Node $v$ exchanges its neighbor set $N(v)$ with all its neighbors;
    Switching node $u$ with two partitioned neighbors is marked TRUE;
    for each node $v \in CDS$ do
        Each node $v$ in $CDS$ exchanges Dominator message to find its dominator ;
        if node $v$ receive Dominator message from node $u$ by normal link $l_{uv}$ then
            Node $v$ sets node $u$ as its Dominator ;
        else
            Node $v$ sets node $u$ as its Dominator, and updates $T_i = T_i + l_{uv}$ ;
    for each node $v \notin CDS$ do
        if node $v$ can not find links in $T_i$ to connect it to another node $u$ in $CDS$ then
            Node $v$ finds switching link $l_{uv}$ to connect it to a node $u$ in $CDS$ ;
            $T_i = T_i + l_{uv}, l_{uv} \in HLS$ ;
    Update $S_i = (G - T_i) \cap HLS$ ;
    Wait for the spectrum handoff of links in $S_i$ to be finished ;
    Update $HLS = HLS - S_i$ ;
    $i++$ ;
```

We explain distributed SHS algorithm by an example in Fig. 4.5(a) and Fig. 4.5(b). In Fig. 4.5(a), the dashed lines denote the switching links. The Fig. 4.5(b) denotes the scheduling results of the distributed SHS algorithm for Fig. 4.5(a). In

Fig. 4.5(b), marked nodes denoted those nodes in CDS, and links marked by T denote the links in reserved spanning tree T_i. We select node 1 as a member of CDS because it is a normal node. We select nodes $5, 6, 7, 8$ as members of CDS because they have two partitioned neighbors. Then we select links to connect them. We first select normal links l_{16},l_{17}, then we select switching links to construct the spanning trees. We can find that if those switching links not in the reserved spanning tree T are removed, the left sub-graph is still connected. This shows that distributed SHS algorithm can guarantee that the network is always connected.



(a) Example of distributed SHS algorithm (b) Result of distributed SHS algorithm

Fig. 4.5: Example of distributed SHS algorithm

The distributed SHS algorithm can be implemented together with an existing routing protocols. After one link predicts that it needs to perform spectrum handoff, it run distributed SHS algorithm to calculate scheduling results. After the ith Switch Link Set S_i is identified, the SHS algorithm will trigger the routing protocol to perform routing update. The links in S_i is assumed to be broken by the routing protocol. The routing protocol performs rerouting to find alternative paths, and switches traffic flows to the new routing path. In this way, the rerouting is performed before spectrum handoff happens.

We analyze the time complexity and feasibility of the distributed SHS algorithm. For time complexity, we prove that the time complexity of the distributed SHS algorithm is polynomial. For feasibility, we prove that the solutions generated by the distributed SHS algorithm can keep the communication of each switching link uninterrupted during spectrum handoff. This shows that the distributed SHS algorithm generates valid solutions.

Theorem6. Given a cognitive network $G(V, E)$ and HLS, the time complexity of distributed SHS algorithm is $O(\alpha(|V|, |E|))$, α is the classical functional inverse of the Ackermann function [Chazelle, 2000].

Proof. Similar to the proof for Theorem 2, the number of iterations for the while loop is $min(|V|, |HLS|)$. In each iteration, one cognitive node spends constant time to find switching link set. Therefore, the total time complexity of the distributed SHS algorithm is $O(\alpha(|V|, |E|))$. $\square$

Theorem7. Given a cognitive network $G(V, E)$ and HLS, the distributed SHS algorithm can keep the communication of each switching link uninterrupted during spectrum handoff.

Proof. We first prove that the distributed SHS algorithm can find scheduling scheme to keep the network connectivity. We prove that phase 1 of SHS algorithm successfully finds a connected dominating set of G. We denote the node set found in [Wu and Li, 1999] by CDS', and that found by the distributed SHS algorithm by CDS. Authors in [Wu and Li, 1999] proved that CDS' is a connected dominating set. Compared with CDS', CDS includes not only nodes in CDS' but also all normal node, so $CDS' \subseteq CDS$. Therefore, CDS is a connected dominating set of G. Therefore, every node in G is connected to at least one node in CDS by links found in phase2 of the distributed SHS algorithm. So sub-graph T_i includes all nodes of G, and it is a connected graph. Because all links in T_i do not perform spectrum handoff, the distributed SHS algorithm can keep the network connectivity during spectrum handoff. Because SHS algorithm can keep the network connectivity, the routing protocol can find alternative paths for each switching link. Therefore, the distributed SHS algorithm can keep the communication of switching links uninterrupted during spectrum handoff. □

4.3.4 A Spectrum-Handoff-aware Rerouting Algorithm

Given the scheduling result of SHS problem, each switching link does rerouting to find an alternative path for the traffic flow on the switching link to keep their communication uninterrupted during spectrum handoff. In this subsection, we design a distributed rerouting algorithm based on $AODV$ routing protocol. The unique requirement for a switch link's alternative path is that it should not find any link that is scheduled to do spectrum handoff in the same round. We explain the detail of the distributed rerouting algorithm as follows.

When a switching link is scheduled to do spectrum handoff in this round, the nodes on the switching link broadcast Routing Request ($RREQ$) messages to find an alternative path. If any node receives the $RREQ$ messages, it checks if the forwarding link needs to do spectrum handoff. If the forwarding link does not need to do spectrum handoff, it forwards the $RREQ$ messages to the next hop. Otherwise, it stops forwarding the $RREQ$ messages. Moreover, the forwarding nodes add some information to the $RREQ$ messages, including its IP information, the left available time and the traffic load of the forwarding link. This information will be used to select alternative paths later. The process is explained by an example in figure .4.6. The dashed lines denote those links which are scheduled to spectrum handoff. Here, link $l(B, C)$ and $l(G, H)$ are scheduled to spectrum handoff in this round, and node B broadcasts $RREQ$ to find alternative path to node D. Node E and F forward $RREQ$ messages to D. Because link $l(G, H)$ also needs to do spectrum handoff, so node G stops forwarding $RREQ$ messages.

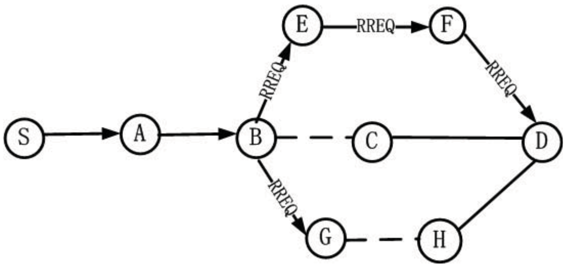


Fig. 4.6: Example of rerouting algorithm.

Because SHS algorithm ensures that the network is always connected, there always exists at least one alternative path for each switching link. If the destination node finally receives multiple $RREQ$ messages, it selects an alternative path using a new routing metric, $LATETX$, which is defined as follows.

Definition 4. **($LATETX$)** Given a link's ETX and left available time (lat), $LATETX$ is defined as following.

$$LATETX(l) = ETX(l)/lat(l) \tag{4.13}$$

We consider two factors here for $LATETX$, the left available time and the expected transmission times (ETX) of each link. ETX denotes the communication condition of each link. The link with more left available time and less ETX should be selected as alternative path. The total $LATETX$ weight of an alternative path is defined as the sum of the $LATETX$ of each link along the alternative path.

$$TotalCost(RP) = \sum_{l_i \in RP} LATETX_{l_i} \tag{4.14}$$

Then, we select the path with the minimum link cost as the alternative path.

$$RP_i = \arg\min_{RP_j}(TotalCost(RP_j)) \tag{4.15}$$

After that, the destination node sends a routing reply messages ($RREP$) to the source node along the selected routing path. The forwarding nodes in the selected routing path forwards $RREP$ to the source node, and the source node switches the traffic to the new alternative path. The detail of the rerouting algorithm is described in Algorithm. .6.

Algorithm 6: REROUTING ALGORITHM

```
Input: $G(V, E)$, Switch link set of the ith round $P_i = \{l_1, l_2, ...|l_i \in R\}$,
       $LATETX$ of each link
Output: Replacing path $P$
for each node $v \in CDS$ do
    Node $s$ broadcasts $RREQ$ to find alternative path to $d$ ;
if node $j$ receives the RREQ and the forwarding link does not need to do
  spectrum handoff then
    Node $j$ forwards $RREQ$ to the next hop ;
else
    Node $j$ stops forwarding;
if node $d$ receives $RREQ$ then
    Node $d$ selects a rerouting path, $RP_i = \arg\min_{RP_j}(TotalCost(RP_j))$;
    Node $d$ sends back $RREP$ to node $d$ along the reverse direction;
Node s receiving $RREP$ switches traffic to the alter-native path ;
```

4.4 Performance Evaluation

4.4.1 Setup of Simulations

In this section, we evaluate the performance of the centralized and distributed *SHS* algorithms by simulations. There are two main goals for our simulations: (i) evaluate the system throughput performance of our *SHS* algorithms while considering impact of number of switching links and traffic flows, and (ii) a comparison of THCT with other solutions while studying impact of number of switching links, left available time and number of radios. We define performance metrics to evaluate the performance of our *SHS* algorithms, including average system throughput, total packet loss ratio in Table.1.

We compare our centralized and distributed *SHS* algorithms with two other solutions, including the solution without any coordination, and the optimal solution. For the solution without any coordination, all switching links perform spectrum handoff simultaneously, and rerouting is performed after spectrum handoff. That is how existing cognitive networks perform multi-link spectrum handoff. The optimal solution is used as an upper-bound to evaluate our *SHS* algorithms. We find the optimal solution by exhaustive search. The optimal solution is not practical because of its high computation cost. We choose simulator Qualnet 4.5 as our simulation tool. We construct a wireless network with 100 wireless nodes, which are randomly distributed. There are totally 18 spectrums from open TV channels [Cordeiro et al., 2006] for all cognitive radios to access. Each spectrum has the same transmission capacity, 2Mbps. We adopt an interference-aware channel assignment algorithm in [Raniwala et al., 2004b] and the standard OLSR routing protocol to assign channels and find routing paths.

4.4.2 Comparison of System Throughput

In this subsection, we evaluate system throughput of the centralized and distributed *SHS* algorithms while considering impact of number of switching links and traffic loads.

a): Impact of number of switching links to system throughput: We now evaluate the system throughput performance of our *SHS* algorithms while considering the impact of number of switching links. In following experiments, each node is equipped with 4 radios, and there are 15 flows in the network. Fig. 4.7 plots the result of system throughput with 10 switching links. We have following observations from Fig. 4.7. First, there is a gap of system throughput caused by spectrum handoff. The system throughput without using *SHS* algorithms is reduced by 30%, while the system throughput of networks using *SHS* algorithms remains is reduced by 10%. This is because *SHS* algorithms can keep the network connectivity, and rerouting can find alternative paths before spectrum handoff really happens. This will keep the communication of switching links uninterrupted during spectrum handoff. Second, the system throughput achieved by our *SHS* algorithms is quite close to that of optimal solution. This optimal solution targets at minimizing the total handoff completion time while keeping the network connectivity rather than always maximizing throughput.Therefore, the Optimal Solution has lower throughput than the Centralized SHS around 9 11 second. Similar phenomenon can also be verified with 100 switching links in Fig. 4.8. In Fig. 4.8, without using *SHS* algorithms, system throughput is reduced by 90% while using *SHS* algorithms, the system throughput is reduced by

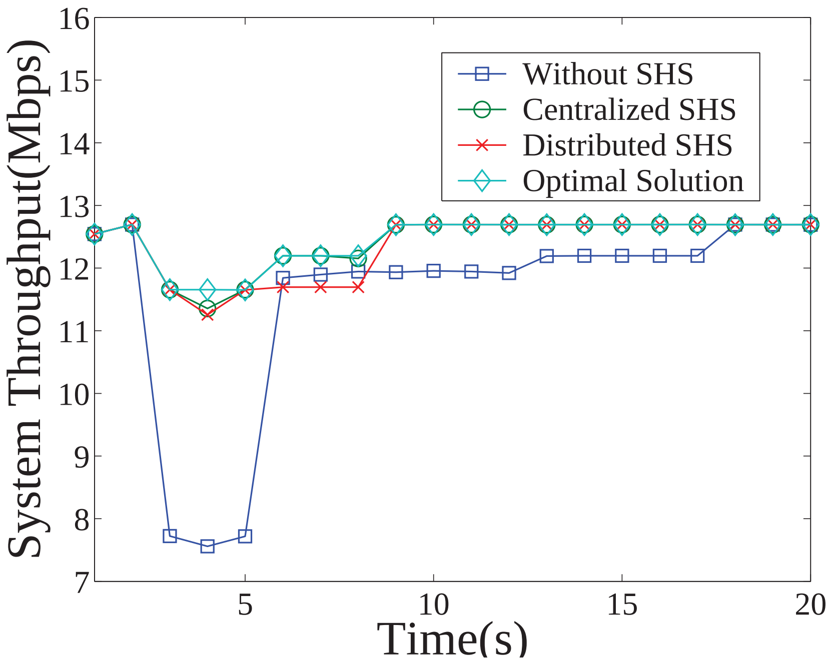


Fig. 4.7: Comparison of system throughput with 10 switching links

about 50%. We test average system throughput of different algorithms with different number of switching links, shown in Fig. 4.9. We can find following phenomenons. First, the network using SHS algorithms can achieve higher system throughput than that without SHS algorithms. This verifies that SHS algorithms can keep the communication of switching links uninterrupted during spectrum handoff. Second, the system throughput decreases as the increase of number of switching links. This is because more links performing spectrum handoff will cause the communication of more links to break. Third, the centralized SHS algorithm performs better than the distributed SHS algorithm. This is because the distributed SHS algorithm calculates the scheduling results based on the local information while the centralized algorithm calculates the scheduling results based on the global information.

b): Impact of traffic loads to system throughput: We now evaluate the

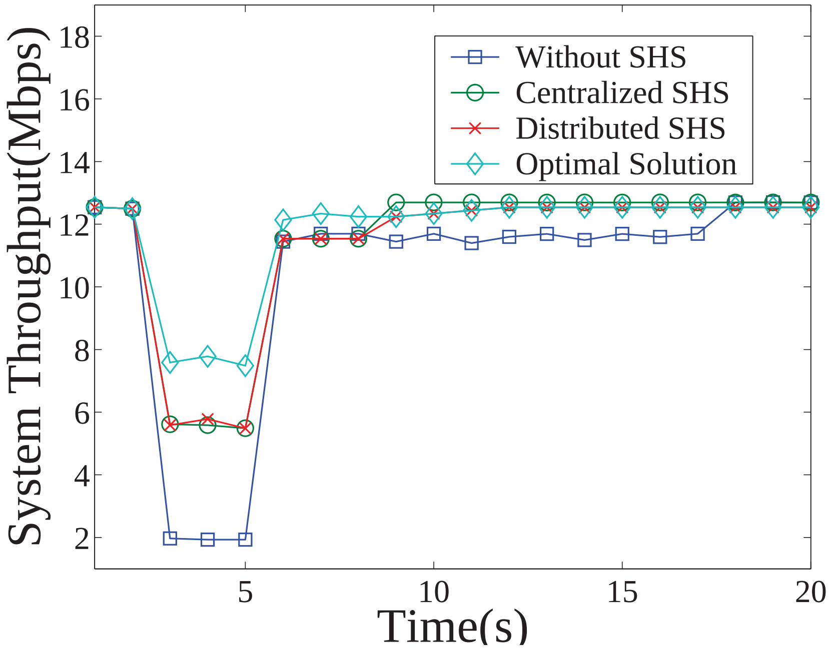


Fig. 4.8: Comparison of system throughput with 100 switching links

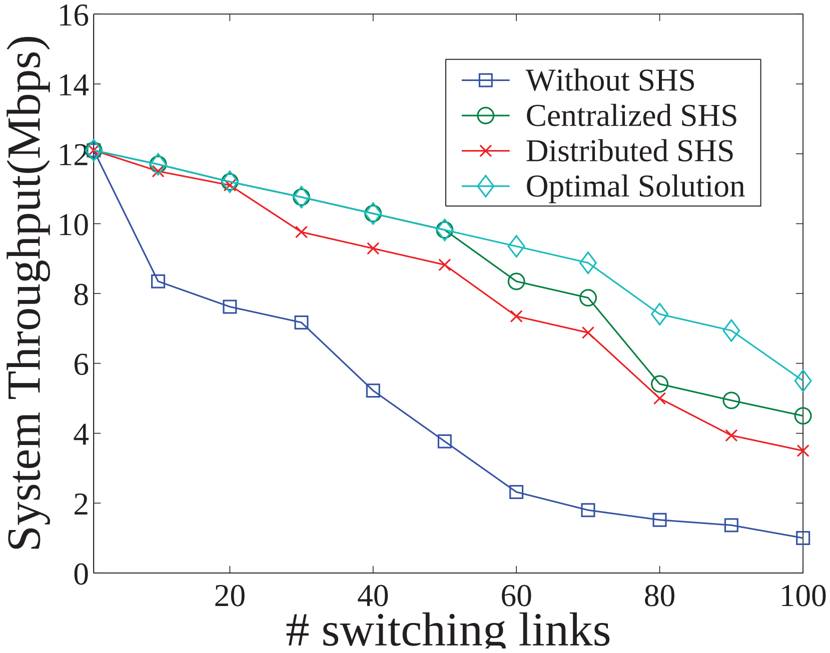


Fig. 4.9: Comparison of average system throughput with different number of switching links

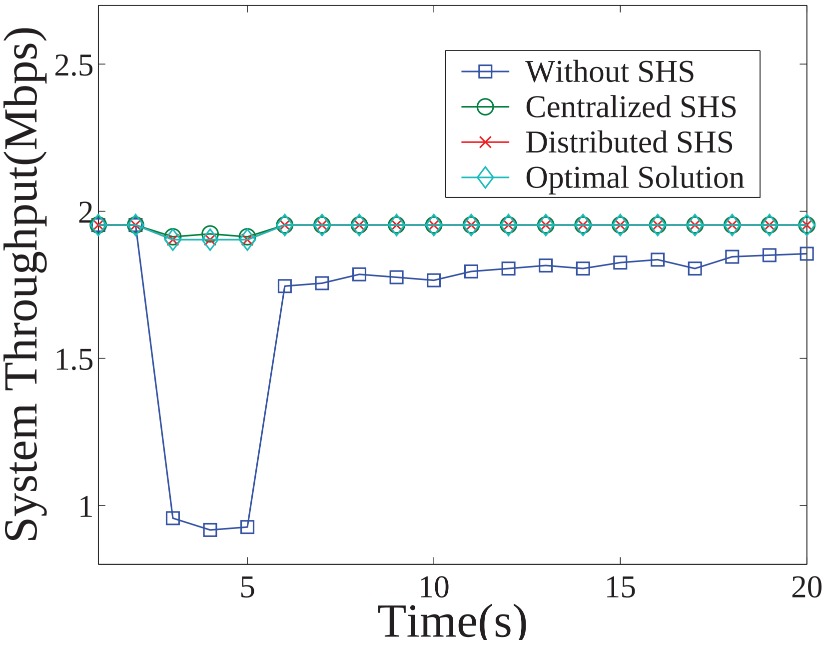


Fig. 4.10: Comparison of system throughput with 5 flows

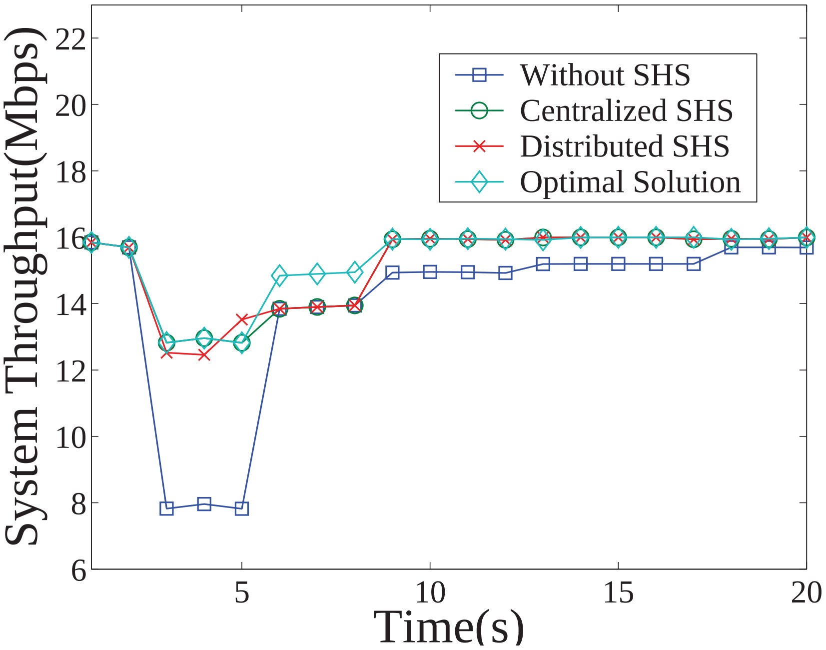


Fig. 4.11: Comparison of system throughput with 20 flows

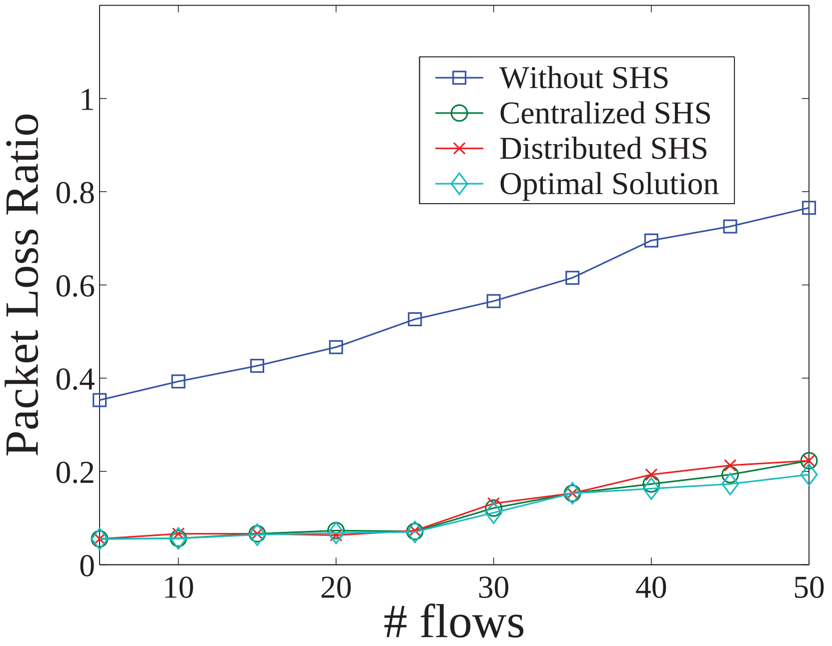


Fig. 4.12: Total packet loss ratio with different number of lows

system throughput performance of our SHS algorithms while considering the impact of traffic loads. In following experiments, each node is equipped with 4 radios, and there are 20 switching links in the network. We test system throughput with 5 TCP flows in Fig. 4.10, and that with 20 TCP flows in Fig. 4.11. We can find that the network using SHS algorithms can achieve higher system throughput than network without using SHS algorithms in both figures. The system throughput does not change too much using SHS algorithms in Fig. 4.10 while the system throughput decreases by 20% using SHS algorithms in Fig. 4.11. Then we try to quantify the impact of traffic load to system throughput by experiments. However, we should not use average system throughput as performance metric because the increase of traffic flows will increase the system throughput even though it causes more degradation of system throughput during spectrum handoff. Therefore, we test the total packet loss ratio during spectrum handoff with different traffic loads shown in Fig. 4.12. We can find following phenomena. First, without using SHS algorithms, the total packet loss ratio increases steadily as the number of traffic flows increases. Using SHS algorithms, the total packet loss ratio is quite low with small number of traffic flows. Second, when the number of traffic flows is larger than a threshold, the total packet loss ratio begins to increase. We analyze the reason for this phenomenon as follows. Without SHS algorithm, the increase of traffic load will cause more traffic flows interrupted during spectrum handoff, which leads to higher packet loss ratio. When we use SHS algorithm, although we switch the traffic flows in switching links to alternative paths, the increased traffic load will cause congestion in the alternative paths, which causes higher packet loss ratio. Third, we can also find that the performance achieved by both centralized and distributed SHS algorithms is quite close to that of optimal solution.

4.4.3 Evaluation of Total Handoff Completion Time

In this subsection, we evaluate THCT of both centralized and distributed SHS algorithms with impact of number of switching links, left available time, and number of radios.

a): Impact of number of switching links to total handoff completion time: We now evaluate the total handoff completition time performance of our SHS algorithms while considering the impact of number of switching links. In following experiments, each node is equipped with 4 radios, and there are 15 flows in the network. We test THCT with different number of switching links shown in Fig. 4.13. We can find following phenomena. First, both centralized and distributed SHS

algorithms can effectively reduce the THCT compared with networks without SHS algorithms. We explain the reason for this phenomena as follows. Without using SHS algorithms, it takes long time for the routing protocol to discover the broken links and trigger routing updates. Moreover, when the routing protocol finds routing paths, it may find routing path with switching links. This causes routing protocol to spend more time in finding routing paths. These problems are well solved by SHS algorithms, and SHS algorithms triggers the rerouting process before spectrum handoff and they will not find alternative routing paths with switching links. This speeds up the process of routing convergence and avoids frequent rerouting. Second, THCT increases as the increases of number of switching links. This is because the increase of number of switching links leads to long scheduling sequence of the SHS algorithms, and it will also take more time for those links to find alternative paths. Third, we can find that the total handoff delay exhibits a step-like behavior w.r.t. the number of switching links. This is because multiple switching links can be scheduled to perform spectrum in single round, increase of switching links will not necessarily increase larger spectrum handoff delay. Fourth, although the performance of SHS algorithm is always very close to the optimal solution, the gap between the optimal solution and our SHS algorithms become large as the number of switching links increases. This is because the increase of number of switching links enlarge the search space of our SHS algorithms, which lead to sub-optimal solution of our SHS algorithms. Finally, we can also find that the total handoff delay of centralized SHS algorithm is less than that of distributed SHS algorithm. This is because the distributed SHS algorithm calculates the scheduling results based on the local information while the centralized algorithm calculates the scheduling results based on

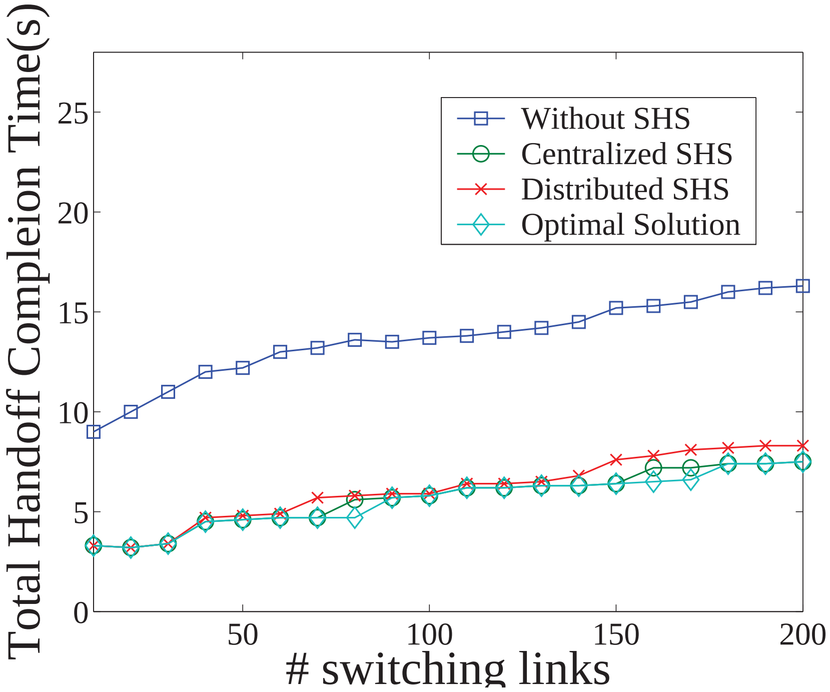


Fig. 4.13: Comparison of total handoff completion time with different number of switching links

the global information.

b): Impact of left available time to total handoff completion time: We now evaluate the THCT performance of our SHS algorithms while considering the impact of left available time. In following experiments, each node is equipped with 4 radios, and there are 20 switching links in the network. There are 15 flows in the network. The testing results of THCT with different left available time are shown in Fig. 4.14. Average left available time here denotes the average time that each can occupy the spectrum before performing spectrum handoff. We have following observations. First, both centralized and distributed SHS algorithms can effectively reduce the THCT with different left available time. This is because SHS algorithms can speed up routing convergence and avoids frequent rerouting. Second, the THCT

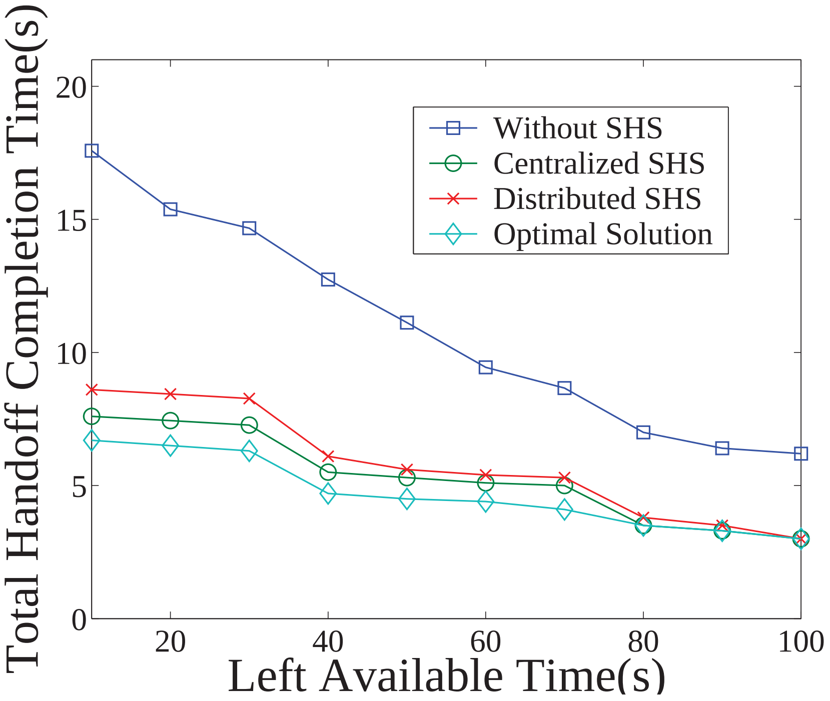


Fig. 4.14: Comparison of total handoff completion time VS left available time

decrease as the increase of left available time. This is because less left available time lead to frequent spectrum handoff, and this will cause more switching links to perform spectrum handoff. Third, both centralized and distributed SHS can achieve performance close to that of optimal solution.

c): Impact of number of radios to total handoff completion time: We now evaluate the THCT performance of our SHS algorithms while considering the impact of number of radios. In following experiments, there are 20 switching links and 15 flows in the network. The testing results of THCT with different number of radios are shown in Fig. 4.15. We can find following phenomena. First, without using SHS algorithms, the number of radios does not have any impact on THCT. This is because the THCT only depends on number of switching links without SHS algorithms. Second, using SHS algorithms, THCT decreases as the increase of number of radios.

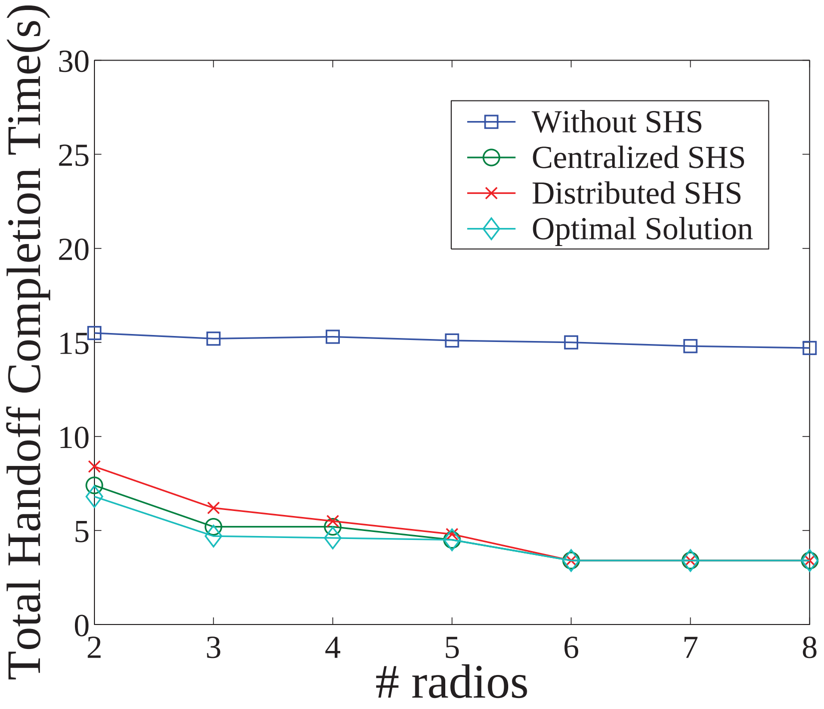


Fig. 4.15: Comparison of total handoff completion time VS number of radios

This is because the increase of radios will increase the network connectivity. As mentioned in Theorem 4 and 5, the increase of network connectivity leads to smaller THCT. Third, the gap between the optimal solution and SHS algorithms decreases as the increase of number of radios. When the number of radios is larger than 5, both centralized and distributed SHS algorithms can achieve the same total THCT as the optimal solution. The verifies the conclusion of Theorem4.

4.5 Conclusion

Spectrum handoff can cause serious performance degradation to the cognitive networks. This paper studies the spectrum handoff problem in the multi-hop scenario which aims at maintaining the network connectivity and minimizing THCT. We propose both centralized and distributed SHS algorithms to solve the SH-MLMH problem. We prove that the centralized SHS algorithm can achieve a logarithmic approximation ratio, and the distributed SHS algorithm can generate valid solutions. Simulation results show that the network performance can be significantly improved by SHS algorithms, in terms of system throughput and THCT. One interesting result shows that THCT achieved by SHS algorithms is related to the number of radios. The THCT achieved by SHS algorithms decreases as the number of radios equipped by each node increases. When the number of radio is larger than a threshold, our SHS algorithms can achieve the same THCT as the optimal solution. The results of this paper can be applied in the future cognitive network, and promote the cognitive radio to real application.

Chapter 5

Non-cooperative Joint Channel and Bandwidth Allocations

5.1 Introduction

Channel allocation is a fundamental issue in wireless networks and has been extensively studied in the recent years. Most of the existing work relied on the assumption of cooperation among different wireless node pairs. However, we can not assume that full cooperation exists in selfish wireless network, so the works based on the assumption of cooperation is not applicable in the selfish wireless network. Existing works have been done for competitive channel allocation in selfish wireless networks. Existing works on competitive multi-radio channel allocation have two major limitations. First, they do not consider the impact of traffic load on the channel transmission quality. Here, traffic load refers to total bandwidth allocated to a channel. They assume that transmission quality of a channel is independent of the channel's traffic load. However, as described in the "Tragedy of the Commons" of

Bandwidth Sharing game (TCBS) in [Nisan et al., 2005], the quality of a single channel will deteriorate as the increase of assigned bandwidth to that channel. This conclusion can be verified in 802.11 CSMA/CA MAC protocol. As we can see from figure .5.1 in [D.Malone and D.Leith, 2007], collision probability increases as the increase of offered load, which verifies that the channel quality deteriorates as the increase of offered load. Moreover, as we can see from figure .5.2 in [Bianchi, 2000], when the offered load is larger than a threshold, congestion will happen and the throughput will decrease as the increase of offered load. These observations remind us that the impact of traffic load to channel quality must be considered when considering bandwidth allocation. We use "Tragedy of the Commons" model in [Nisan et al., 2005] to describe this impact in the paper. Second, existing works assumed that each node pair always has packets to send, implying saturated packet arrival rates, which is an extreme case in practice. The packet arrival rate is often limited in the real situation. Moreover, depending on applications, packet arrival rates in links of different node pairs might be quite different. For example, packet arrival rates of VOIP and online video streaming are quite different [B.Davie et al., 2002]. Competitive multi-radio channel allocation solution [Alicherry et al., 2005] based on this assumption of saturated packet arrival rates may lead to low channel utilization efficiency. For example, some channels might be congested while others might be idle. Therefore, we should assume that each node pair has limited packet arrival rates, and packet arrival rates of different node pairs can be different.

In designing new solutions under the new assumptions, we extend the problem of non-cooperative multi-radio channel allocation to non-cooperative joint channel and bandwidth allocation problem, referred to as NJCBA problem. In the NJCBA

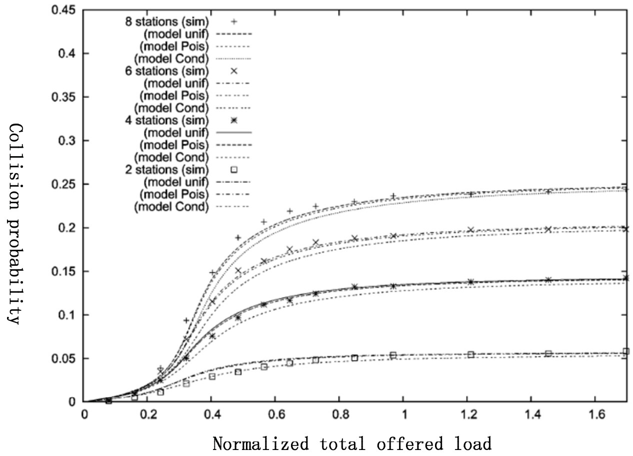


Fig. 5.1: Relationship between collision probability and offered load.

problem, node pairs compete for both channel and bandwidth resource to maximize their utility, which is related to its data rate and transmission quality. We adopt the approach of game theory to solve the NJCBA problem. We model the problem as a non-cooperative static game, called joint channel and bandwidth allocation game, referred to as NJCBA game. We try to answer three important problems for the NJCBA game: What is the stable state of the NJCBA game? How about the system efficiency of the stable state? How to design algorithms to enable players to converge to the stable state quickly?

Using the knowledge from game theory, we usually assume that Nash Equilibrium is the final stable state of a non-cooperative game. Moreover, game theory defines two kinds of NEs of a given game, mixed-strategy NE and pure NE. In pure NE, each player can only take one identical strategy while in mixed-strategy NE, each player takes the mixed strategies, where players choose a probability distribution over possible actions. The mixed-strategy NEs are not always the stable state of a game, while the pure NEs are always assumed as the stable state of a game. Nash

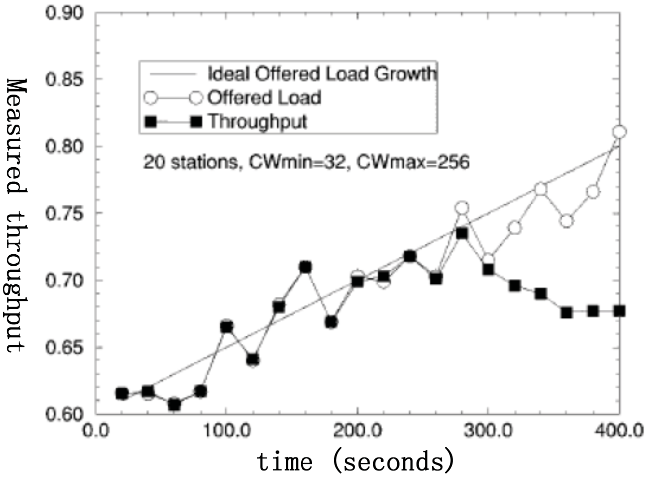


Fig. 5.2: Measured Throughput with slowly increasing offered load.

proved that if we allow mixed strategies, then every n-player game admits at least one mixed-strategy Nash Equilibrium. However, a general game does not necessarily have pure NEs, so we need to analyze if the NJCBA game have pure NEs or not. Authors in [G.Gottlob et al., 2005] proved that determining whether a general game has a pure Nash Equilibrium is NP-hard, so it is a very challenging to prove that a game has pure NEs. In this paper, we prove that the NJCBA game has pure NEs, which shows that the NJCBA game can converge to the final stable state using pure NEs. We also analyze the system efficiency of the NEs state by the concept of Price Of Anarchy (POA). We prove that in the heavy-load network, NEs of NJCBA can achieve a constant POA. This shows that even if every player adopts selfish strategies, the network can still achieve a guaranteed performance. We design two distributed polynomial algorithms to enable players to converge to an NE with different assumption on available information. The first algorithm is to enable players converge to an NE in the case that each player has perfect information about the allocation result. The second algorithm is to enable players to converge to an NE in the case that each player has imperfect information about the allocation result. Imperfect information here refers to the case that each player only knows the channel and bandwidth allocation result of those channels on which the player has radio to operate. We evaluate the performance of our two algorithms through extensive simulations. We find that our two algorithms can improve system throughput by 2 to 3 times compared with solutions in work [Alicherry et al., 2005] at the stable state, and our algorithms can converge to the stable state at a fast speed.

The rest of the paper is organized as follows. We briefly review the related works in Section II and present the technical preliminaries in Section III. In Section IV, we prove the existence of pure NEs for joint allocation game using the best response concept. We design two distributed algorithms to enable players converge to NEs in Section V. And we present the evaluation results in Section VI. Finally, we conclude the paper and point out potential future works in Section VII.

5.2 Modeling of Competitive Allocation Problem

5.2.1 Network Model

We assume that the available frequency band is divided into multiple orthogonal channels of the same bandwidth using OFDM technology. The set of available channels is denoted as $C = \{c_1, c_2, ..., c_N\}$. Each device is equipped with $k < |C|$ radio transmitters, all having the same communication capability. We denote the capacity of a channel by M. In our model, pairs of nodes need to communicate with each other over a single hop. We assume that each user participates in only one such communication session. We assume that there is a mechanism that enables the players to use multiple channels to communicate simultaneously. We assume that $|N| * k > |C|$, hence the devices have a conflict during the channel allocation process. For node pairs working on one channel, we assume that CSMA/CA MAC protocol is adopted to coordinate their access to the channel. We assume that all node pairs exist in a single collision domain, which means that each device can hear the transmissions of every other device if they are using the same channel. We denote the bandwidth demand of node pair i by R_i.

5.2.2 Theoretical game model

We model the $NJCBA$ problem as a non-cooperative static game, referred to as $NJCBA$ game, $G(N, S, U)$. Here, N denotes a set of players, S denotes the strategy space of each player, U denotes the utility function of each player. We refer to each node pair as a selfish player in N. We define the strategy of player i and its constraints as follows.

$$s_i = \{(k_{i,c_1}, t_{i,c_1}), ..., (k_{i,c_{|C|}}, t_{i,c_{|C|}})\} \tag{5.1}$$

Subject to

$$if t_{i,c} > 0, k_{i,c} = 1; \tag{5.2}$$

$$\sum\nolimits_{c} k_{i,c} \leq k; \tag{5.3}$$

$$t_{i,c} \geq 0, \sum_{c \in C_i} t_{i,c} \leq R_i; \tag{5.4}$$

Here, $k_{i,c}$ denotes the number of radios that player i allocates to channel c , and $t_{i,c}$ denotes the bandwidth that player i allocates to channel c. We find three constraints that the BR strategy of player i must satisfy. Constraint (5.2) denotes that if one node allocates bandwidth to a channel, then it must allocates one radio to that channel; constraint (5.3) denotes totally assigned radios must be less than its total number of radios; Constraint (5.4) denotes that each player's totally allocated bandwidth must be no more than its bandwidth demand. We denote the set of channels selected by player i by C_i , $C_i \subset C$, called working channel set of player i. We denote the total number of radios allocated to channels c by all players by k_c. We denote the traffic load on channel c as T_c, $T_c = \sum\limits_{j} t_{j,c}$. We denote the traffic load of channel c except the bandwidth allocated by player i as $T_{-i,c}$, $T_{-i,c} = \sum\limits_{j \neq i} t_{j,c}$. The strategy vectors of all players define the strategy matrix S, where the row i of the matrix corresponds to the strategy vector of player i.

$$S = \begin{pmatrix} s_1 \\ ... \\ s_{|N|} \end{pmatrix} \tag{5.5}$$

We denote the strategy matrix except for the strategy of player i by S_{-i}. If player i allocates bandwidth to channel c, we use the Tragedy of the Commons model about bandwidth-sharing in [Gao and Wang, 2008] to define the utility that player i get from channel c as following.

$$u_{i,c} = \begin{cases} t_{i,c} * (1 - \sum\limits_{j} t_{j,c}/M), if \sum\limits_{j} t_{j,c} < M \\ 0, if \sum\limits_{j} t_{j,c} \geq M \end{cases} \tag{5.6}$$

Here, $\sum_j t_{j,c}$ denotes total traffic load on channel c, $\sum_j t_{j,c}/M$ denotes the normalized traffic load on channel c, $(1 - \sum_j t_{j,c}/M)$ denotes the quality of channel c, which deteriorates as the increase of traffic load. The quality of a channel represents the benefits that singe player gets by transmitting unit packets in that channel. When the traffic load on channel c is less than its capacity, the utility of player i is defined as the benefits by allocating $t_{i,c}$ to channel c. When the traffic load on channel c is larger than its capacity, we assume that no player can get benefit from channel c. Although this utility function does not precisely describe mathematical relationship between channel quality and assigned bandwidth, it reflects the trend of channel quality when traffic load is increasing. We explain this by two examples. As in figure .5.1 from [Das et al., 2005] , the collision probability increases as the increase of offered load, which means that the quality of a channel deteriorates as the increase of its traffic load. As in figure .5.2 from [Fudenberg and Tirole, 1991], when the offered load is larger than a threshold, congestion will happen and the throughput will decrease as the increase of offered load. In this case, we assume that the transmission quality of the channel is not tolerable, so no player will get any benefit from that channel. We define the utility function of player i as the sum of utility from all selected channels, $U_i = \sum_{c \in C_i} u_{i,c}$.

5.2.3 Notations and concepts in game theory

We explain some concepts about game theory, including Nash Equilibrium and the best response strategy in [Raniwala and T.Chiueh, 2005]. In an NE strategy profile, none of the players can unilaterally change its strategy to increase its payoff, so NE is often considered to be the stable state of a game. The Best Response

(BR) strategy of player i is defined as: given other's players' allocation result, the strategy which can produce the maximum benefit for player i. We explain relationship between the BR strategy and NE. In an allocation scheme, if each player adopts the BR strategy, that allocation scheme is NE scheme. This reminds us that we can prove the existence of NEs of the $NJCBA$ game using the best response concept.

Definition 1: (Nash Equilibrium, NE) The strategy matrix $S^* = \{s_1^*, ..., s_{|N|}^*\}$ defines a Nash Equilibrium (NE), if for every player i, we have $U_i(s_i^*, S_{-i}^*\} \geq U_i(s_i^{'}, S_{-i}^*\}$, for every strategy $s_i^{'}$.

Definition 2: (Best Response strategy, BR) The best response strategy $B_i(S_{-i}^*)$ of player i is defined as the strategy s_i^* which produces the most favorable outcome for player i, if other players' strategies S_{-i}^* are given.

Theorem1. In a normal-form game, a combination of strategies $\{s_1^*, ..., s_n^*\}$ is a Nash Equilibrium if for every player i, $s_i^* \in B_i(S_{-i}^*)$.

5.3 Analysis and Algorithm Design

5.3.1 Existence of Pure NEs

In this subsection, we prove the existence of pure NEs for NJCBA game. We explain the reason to prove the existence of pure NEs as follows. Game theory defines two kinds of NEs, mixed-strategy NE and pure NE. In pure NE, each player takes one identical strategy while in mixed-strategy NE, each player takes the mixed strategies, where players choose a probability distribution over possible actions. The mixed-strategy NEs are not always the stable state, while the pure NEs are always the stable state. Nash proved that every n-player game admits at least one mixed-strategy NE in [Raniwala and T.Chiueh, 2005]. However, a general game does not necessarily have pure NEs, so we need to analyze if the NJCBA game have pure NEs or not. Authors in [D.Malone and D.Leith, 2007] proved that determining whether a general game has a pure NE is NP-hard, so it is challenging to prove that a game has pure NEs. From theorem2, we find that BR strategy has close relationship with the NEs. So we first find each player's BR strategy by solving an optimization problem. Then we prove the existence of pure NEs by analyzing the structure of the BR strategy. Given other players' allocation result, player i's BR strategy of the NJCBA game should be the optimal solution to an optimization problem, denoted by Best Response problem (BR) .

$$MaximizeU_i = \arg\max \sum_{c \in C_i} t_{i,c}(M - \sum_j t_{j,c}) \tag{5.7}$$

Subject to (5.2),(5.3),(5.4).

We find player i's best response strategy in three steps. First, we find player i's working channel set C_i . Second, we find player i's bandwidth allocation result in BR strategy without bandwidth demand constraints. Third, we find player i's best bandwidth allocation result with bandwidth demand constraints. By combining these three results, we can find player i's overall best response strategy. First, we find player i's working channel set C_i. Because a channel's transmission quality becomes worse as increase of traffic load working on it, player i should select k top channels with least traffic load as his working channel set, C_i. Then, we can get working channel set $C_i = \{c | \forall c^{'} \notin C_i, T_{-i,c} \leq T_{-i,c'}, |C_i| = k\}$. Second, we find the bandwidth allocation of player i in working channels c without bandwidth demand constraint. We denote the bandwidth allocated by player i to channel c without bandwidth demand constraints by $t_{i,c}^D$, and total allocated bandwidth without demand constraint by T_i^D, $T_i^D = \sum\limits_{c \in C_i} t_{i,c}^D$. We find optimal solution for BR problem without demand constraint:

$$\partial Utility_i / \partial t_{i,c} = t_{i,c}^D = (M - T_{-i,c})/2, \forall c \in C_i \tag{5.8}$$

Third, we solve the BR problem with bandwidth demand constraint. We represent total bandwidth allocated by player i in the BR strategy as T_i . We can find that $T_i = \min\{T_i^D, R_i\}$. We can further find that

$$ifT_i^D \le R_i, T_i = T_i^D, t_{i,c} = t_{i,c}^D \tag{5.9}$$

$$ifT_i^D > R_i, T_i = \sum_{c \in C_i} t_{i,c} = R_i \tag{5.10}$$

Because case 1 has been analyzed in the second step, we focus on case 2 in the third step. We first construct Lagrange multipliers for the BR problem.

$$L_i(t_{i,c_1}, t_{i,c2}, ..., \lambda_i) = U_i + \lambda_i * (R_i - \sum_{c \in C_i} t_{i,c}) \tag{5.11}$$

We list $K-T$ conditions in [Alicherry et al., 2005] for the BR problem.

$$\begin{cases} t_{i,c} * \partial L_i(t_{i,c_1}, t_{i,c_2}, ..., \lambda)/\partial t_{i,c} = 0 \\ \partial L_i(t_{i,c_1}, t_{i,c_2}, ..., \lambda)/\partial t_{i,c} \leq 0 \\ \sum\limits_{c \in C_i} t_{i,c} \leq R_i \\ \lambda_i(\sum\limits_{c \in C_i} t_{i,c} - R_i) = 0 \\ \lambda_i \geq 0 t_{i,c} \geq 0 \end{cases} \tag{5.12}$$

By analyzing K-T conditions, we can find lemma 1. Here, case1 denotes that if the traffic load in channel c_2 is larger than that in channel c_1 over a threshold, then player i will not allocate any bandwidth to channel c_2. Case2 denotes that if node i allocates bandwidth to both channel c_1,c_2, the difference between other players' traffic load in channel is related to the difference of player i's allocated bandwidth in channel c_1,c_2. Based on Lemma1, we prove the existence of pure NEs for $NJCBA$ game in theorem2.

Lemma 1: Player i's best response strategy $s_i^* \in B_i(S_{-i}^*)$ must satisfy following conditions: $\forall c_1, c_2 \in C_i$, Case1: If $t_{i,c_1} > 0$, $T_{-i,c_2} - T_{-i,c_1} \geq 2 * t_{i,c_1}$, then $t_{i,c_2} = 0$; Case2: If $t_{i,c_1} > 0, t_{i,c_2} > 0$, then $T_{-i,c_1} - T_{-i,c_2} = 2(t_{i,c_2} - t_{i,c_1})$.

Proof: $\forall c_1, c_2 \in C_i$, if $t_{i,c_1} > 0, t_{i,c_2} = 0$, by equation (12) and (13), we can get

$$\partial L_i(t_{i,c_1}, t_{i,c_2}, ..., \lambda_i)/\partial t_{i,c_1} \geq \partial L_i(t_{i,c_1}, t_{i,c_2}, ..., \lambda_i)/\partial t_{i,c_2} \tag{5.13}$$

$$\partial L_i(t_{i,c_1}, t_{i,c_2}, ..., \lambda)/\partial t_{i,c} = M - T_{-i,c_1} - 2t_{i,c} - \lambda_i \tag{5.14}$$

By permutation, we can get

$$T_{-i,c_2} - T_{-i,c_1} \geq 2 * t_{i,c_1} \tag{5.15}$$

Therefore, Case 1 is proven. If $t_{i,c_1} > 0, t_{i,c_2} > 0$, by equation (12), we can get

$$\partial L_i(t_{i,c_1}, t_{i,c_2}, ..., \lambda_i)/\partial t_{i,c_1} = \partial L_i(t_{i,c_1}, t_{i,c_2}, ..., \lambda_i)/\partial t_{i,c_2} \tag{5.16}$$

By permutation, we can get

$$T_{-i,c_1} - T_{-i,c_2} = 2(t_{i,c_2} - t_{i,c_1}) \tag{5.17}$$

Therefore, Lemma1 is proven. □

Theorem 2. There exist pure NEs for the NJCBA game.

Proof. According to property in case1 of lemma1, we find those selected channels, which we do not allocate any bandwidth. Suppose that player i does not allocate bandwidth to selected channel c, following relationships must be satisfied between bandwidth allocated to channel c and other selected channels .

$$\begin{cases} T_{-i,c} - T_{-i,c_1} \geq 2 * t_{i,c_1} \\ T_{-i,c} - T_{-i,c_2} \geq 2 * t_{i,c_2} \\ ... \\ T_{-i,c} - T_{-i,c_{|C_i|}} \geq 2 * t_{i,c_{|C_i|}} \end{cases} \tag{5.18}$$

By solving the equations (5.10) and (5.18), we can find:

$$if(k-1)*T_{-i,c}-\sum_{c'\neq c}T_{-i,c'}\geq 2t_i, t_{i,c}=0 \tag{5.19}$$

We denote sub-set of working channels which node i does not allocate bandwidth by $C_i^0=\{c|c\in C_i, t_{i,c}=0\}$. According to property in case2 of lemma1, we allocate bandwidth to left channel c. According to property in case 2 of Lemma 1, for channel c and c_j, $c_j\in C_i-C_i^0, c_j\neq c$, it satisfies following equations.

$$\begin{cases} T_{-i,c}-T_{-i,c_1}=2(t_{i,c_1}-t_{i,c}) \\ T_{-i,c}-T_{-i,c_2}=2(t_{i,c_2}-t_{i,c}) \\ \dots \\ T_{-i,c}-T_{-i,c_{|C_i|}}=2(t_{i,c_{|C_i|}}-t_{i,c}) \end{cases} \tag{5.20}$$

By solving the equations (5.10) and (5.20), we can get

$$t_{i,c}=\frac{2*R_i-\sum\limits_{c'\neq c}(T_{-i,c}-T_{-i,c'})}{2*(|C_i|-|C_i^0|)} \tag{5.21}$$

By combining results in step 2 and 3, the bandwidth allocation vector in the BR strategy of player i can be found.

$$t_{i,c}=\begin{cases} (M-T_{-i,c})/2, c\in C_i-C_i^0, if \sum\limits_{C_i-C_i^0}(M-T_{-i,c})/2<R_i \\ \frac{2*R_i-\sum\limits_{c'\neq c}(T_{-i,c}-T_{-i,c'})}{2*(|C_i|-|C_i^0|)}, c\in C_i-C_i^0,, if \sum\limits_{C_i-C_i^0}(M-T_{-i,c})/2\geq R_i \\ 0, c\in C-C_i, or c\in C_i^0 \end{cases} \tag{5.22}$$

Because we find that the best response strategy is continuous and bounded, according to Brouwer's fixed point theorem in [Raniwala and T.Chiueh, 2005], we can conclude that there always exist pure NEs for the NJCBA game. □

5.3.2 Price of anarchy for pure NEs

We analyze the efficiency of pure NEs for the $NJCBA$ game using the concept of POA. We prove that in the heavy-loaded network, pure NEs can achieve a constant POA.

Theorem3. In a NJCBA game, if $\forall i \in N$, $R_i \geq k * M/2$, pure NEs of NJCBA can achieve a POA of $([k * |N|/C] + 1)/4$.

Proof. We find the upper bound of the globally optimal solution for $NJCBA$ game, whose objective is to maximize sum of utilities of all players.

$$Maximize(U) = \sum_{i \in N} U_i = \sum_{c \in C} T_c * (M - T_c)/M \tag{5.23}$$

Subject to constraints (5.2),(5.3),(5.4). We find the upper bound of U with mathematical knowledge,

$$U \leq |C| * M/4 \tag{5.24}$$

We find the bandwidth allocated to each channel c by player i, $t_{i,c}^D = (M - T_{-i,c})/2$. We find it satisfies the bandwidth demand constraint because $T_i^D = \sum_{c \in C_i} t_{i,c}^D \leq k * M/2$.

Therefore, we find that $t_{i,c} = t^D_{i,c}$, which holds for every player working on channel c.

$$t_{i,c} = \begin{cases} t_{1,c} = (M - T_{-1,c})/2 \\ t_{2,c} = (M - T_{-2,c})/2 \\ \ldots\ldots \\ t_{i,c} = (M - T_{-i,c})/2 \end{cases} \tag{5.25}$$

By solving the equitation set, we can find the BR bandwidth allocation result.

$$t_{i,c} = M/(k_c + 1) \tag{5.26}$$

$$U_{i,c} = \frac{M}{(k_c + 1)^2} \tag{5.27}$$

We can find that $\forall c_1, c_2 \in C, k_{c_1} - k_{c_2} \leq 1$, because otherwise, user i working on channel c_1 not on channel c_2 will switch its radio to channel c_2 to improve its utility. So we can find that:

$$kc = [k * |N|/C] or k_c = [k * |N|/C] + 1 \tag{5.28}$$

We calculate total utility of NE allocation as follows.

$$U_{NE} = \sum_{c \in C} U_{i,c} = \sum_c \frac{k_c * M}{(k_c + 1)^2} \approx \sum_c \frac{M}{k_c + 1} \geq \frac{C * M}{[k * |N|/C] + 1} \tag{5.29}$$

Here, operator $[*]$ refers to the operation that takes integral part of $*$, when k_c is large enough, $\frac{k_c * M}{(k_c+1)^2} \approx \frac{M}{k_c+1}$. We can prove POA of NE allocation .

$$POA = U_{\max}/U_{NE} \geq ([k * |N|/C] + 1)/4 \tag{5.30}$$

Therefore, theorem3 is proven. □

5.3.3 Algorithm Design

In this subsection, we design algorithms to enable players to converge to NE. Given an initial random allocation, every player adopts the best response strategy in a distributed way to maximize its own benefit. According to Theorem 2, the system will converge to pure NE stable state finally. We face two new challenges when designing distributed algorithms. The first challenge is the unstable channel allocations caused by simultaneously moving of different players. We solve this challenge by a back-off mechanism. The second challenge is the imperfect information in the algorithm with perfect information. We solve this challenge by an opportunistic access mechanism.

5.3.4 Distributed algorithm with perfect information

We design a distributed algorithm to enable players to converge to one NE allocation scheme. First, we assume that there exists a random-based channel and bandwidth allocation over channels. Then to avoid the unstable channel allocations caused by simultaneously moving of different players, we use the technique of back-off mechanism well known in the IEEE 802.11 medium access technology. We denote the back-off window by W and each coalition chooses a random initial value for his back-off counter with uniform probability from number set $\{1, ..., W\}$.

When back-off counter of one player becomes 0, he uses best-response function to find new strategy to get more benefit. We set a threshold ξ for a player to decide whether to take new strategy. If the difference between utility of a player after and before changing his allocation strategy is larger than ξ, the player takes new strategy. Otherwise, the player will not take new strategy. If finally no players change their strategies, we assume that system reaches stable state.

Algorithm 7: DISTRIBUTED ALGORITHM WITH PERFECT INFORMATION

```
Input: Channel set $C = \{c_1, c_2, ..., c_N\}$, channel capacity $W$, each channel's
       traffic load $T = T_{c_1}, T_{c_2}, ..., T_{c_N}$
Output: Allocation vector $s_i = \{(k_{i,c_1}, t_{i,c_1}), ..., (k_{i,c_{|C|}}, t_{i,c_{|C|}})\}$
while do
    for player i from 1 to N do
        if back-off counter is 0 then
            Select $k$ channels with least traffic load as working channel set $C_i$;
            Calculate bandwidth allocation vector $s_i^{'}$ using best response
            strategy ;
            if $U_i^{'}(s_i^{'}, S_{-i}^{'}) - U_i(s_i, S_{-i}^{'}) > \xi$ then
                $s_i = s_i^{'}$, $U_i(s_i, S_{-i}^{'}) = U_i^{'}(s_i^{'}, S_{-i}^{'})$ ;
            else
                Do nothing ;
            Restore back-off counter ;
        else
            Decrease back-off counter value by one ;
```

5.3.5 Distributed algorithm with imperfect information

The distributed algorithm described above requires perfect information, which is not practical in the real selfish wireless network. We assume that players have imperfect information, meaning that they know the channel and bandwidth allocation result on only those channels on which they operate a radio. Based on this assumption, we design a distributed algorithm to enable players to converge to the stable state, called convergence algorithm2.

The major difference between algorithms with imperfect information and perfect information is the opportunistic mechanisms we introduced for channel allocation, and they proceed as follows. Similar to algorithm1, an initial random channel and bandwidth allocation is assumed, and back-off mechanism is also taken here. When player i's back-off counter reaches 0, he checks if there exist channel $c \in C_i$ with $t_{i,c} = 0$. If there exists channel $c \in C_i$ with $t_{i,c} = 0$, player i moves his radio to another channel $c \notin C_i$ with a probability. The probability to choose a channel $c \notin C_i$ is $\frac{1}{|C-C_i|}$. If every channel $c \in C_i$, $t_{i,c} > 0$, player i selects channel $b \in C_i$ with minimum allocated bandwidth, and with small probability p, player i moves his radio to another channel $c \notin C_i$. After that, player i calculates bandwidth allocation vector using best response algorithm. Although the two opportunistic mechanisms introduced here solve the problem of imperfect information, it causes the instability of NE.

5.4 Performance Evaluation

In this section, we evaluate the performance of our two algorithms by simulations. Several aspects are presented including convergence speed, system performance of the stable state. We implement our two distributed algorithms in Qualnet. We use IEEE 802.11a as our communication protocol. We assume that the number of orthogonal channels is 12, $|C| = 12$. The number of radios each player is equipped with is 3. The number of node pairs is 50, $|N| = 50$. The capacity of each channel is

Algorithm 8: DISTRIBUTED ALGORITHM WITH IMPERFECT INFORMATION

```
Input: Channel set $C = \{c_1, c_2, ..., c_N\}$, channel capacity $W$, each channel's traffic load $T_{c_i}$
Output: Allocation vector of player $i$ $s_i = \{(k_{i,c_1}, t_{i,c_1}), ..., (k_{i,c_{|C|}}, t_{i,c_{|C|}})\}$
Random channel and bandwidth allocation ;
while do
    for player $i$ from 1 to $N$ do
        if back-off counter is 0 then
            for Radio $j$ from 1 to $k$ do
                Suppose radio $j$ works on channel $b$ ;
                if $t_{i,b} = 0$ then
                    Player $i$ moves the radio $j$ from $b$ to $c \notin C_i$, where $c$ is chosen with uniform random probability from the set $C - C_i$;
                else
                    if $t_{i,b} = \arg\min_c(t_{i,c})$ then
                        Player $i$ moves the radio $j$ from $b$ to $c \notin C_i$ with probability p, where $c$ is chosen with uniform random probability from the set $C - C_i$;
                        Calculate bandwidth allocation result using best response strategy;
                        Restore back-off counter ;
        else
            Decrease back-off counter value by one ;
```

$12Mbps$. The transmission range is set to $235m$, and distance between any two nodes is within that range. The range for the bandwidth demand of node pairs varies from 20Kbps to 2Mbps. We denote the distributed algorithm with perfect and imperfect information by convergence algorithm1 and convergence algorithm2.

5.4.1 Evaluation of Convergence Speed

We evaluate the convergence speed of our algorithms. The convergence speed is measured by the number of rounds that the system takes to converge to the stable state (NE). Rounds here refer to how many times that players change their allocation strategies.

Due to instability of convergence algorithm2 caused by imperfect information, we define convergence time of convergence algorithm2 as the first time that allocation scheme generated by convergence algorithm2 reaches NE states. In figure .5.3, it takes algorithm1 8 rounds to converge to an NE allocation while it takes convergence algorithm2 26 rounds to converge to an NE allocation. In figure .5.4, we compare the convergence time of two algorithms with different traffic profiles. Traffic profile here refers to the distribution of all players' packet arrival rate. In .5.4, convergence algorithm2 takes about 24 rounds while convergence algorithm1 takes about 7 rounds to reach the stable state. We conclude that both algorithms converge to the stable state at a fast speed, and algorithm1 converges to the states faster.

5.4.2 Evaluation of System Performance

In this subsection, we compare the system performance of stable state (NE) converged by our two algorithms and the first kind of NEs in [Alicherry et al., 2005].

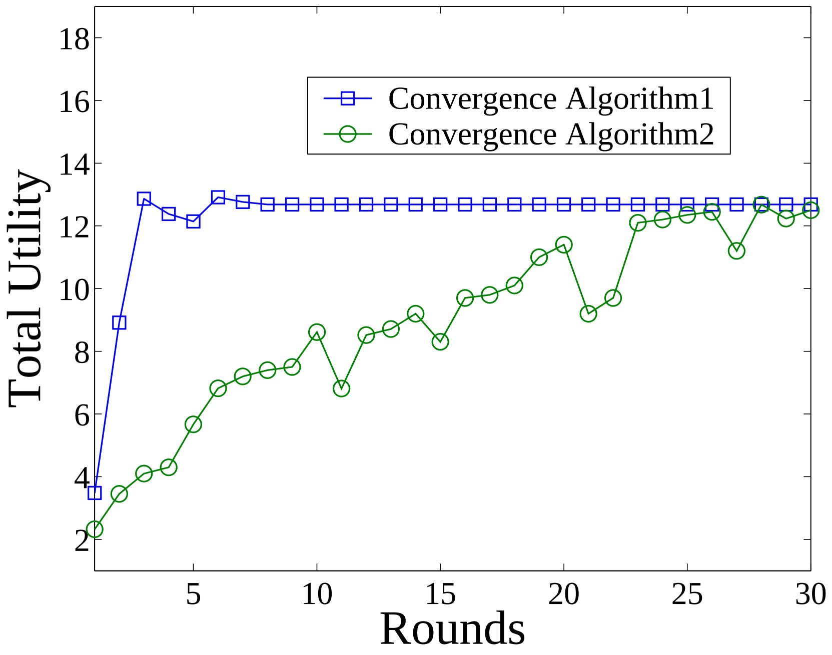


Fig. 5.3: Convergence Process Analysis.

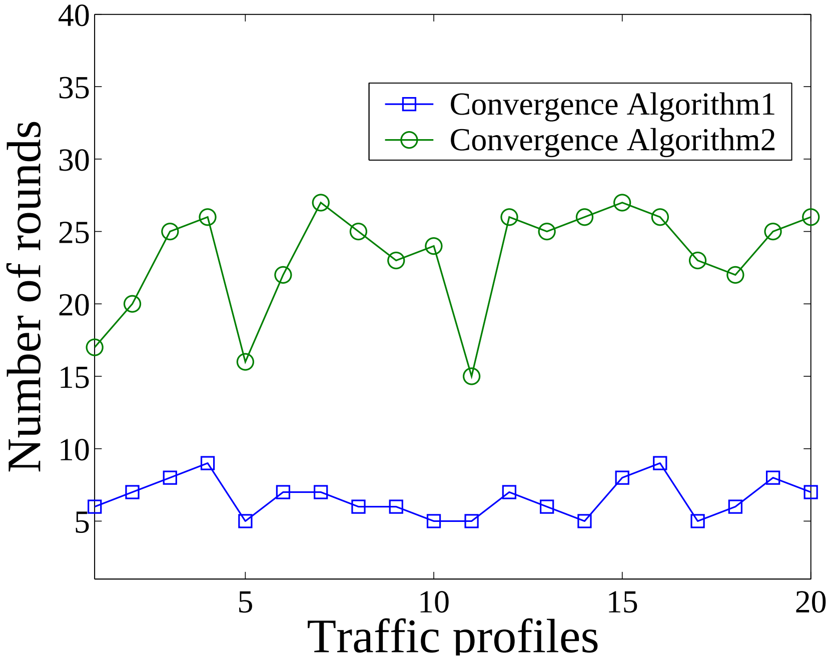


Fig. 5.4: Convergence speed comparison.

The measurement metrics include total utility, system throughput and packet loss. As in works [Alicherry et al., 2005], we assume that bandwidth is allocated equally among all selected channels. What's more, because the instability of algorithm2, we measure its system performance by the average total utility, average system throughput, average packet loss ratio. Here, after algorithm2 converges to NE, we continue recording related values of 1000 rounds, and take the average values as our results. The related metrics are defined as following.

About total utility, as we can see from figure .5.5, our two algorithms improve the utility values by about 2 times compared with NEs achieved in works [Alicherry et al., 2005]. What's more, the results in figure .5.6 show that stable states converged by our two algorithms improves the system throughput 2 or 3 times compared with works [Alicherry et al., 2005]. About packet loss ratio, from figure .5.7 we find that the packet loss ratio of stable states converged by our two algorithms is about 1/4 of packet loss ratio of NEs in works [Alicherry et al., 2005]. Finally, we compare load-balancing property of stable states converged by our two algorithms and that of NEs in works [Alicherry et al., 2005]. The results in figure .5.8 show that using stable allocation converged by our two algorithms, traffic loads are generally equally allocated over all channels. However, using NE allocation scheme, the traffic loads over different channels are quietly different. This result shows that our algorithms can utilize the channel and bandwidth resource more efficiently than works [Alicherry et al., 2005].It's understandable why our algorithms improve system performance significantly. The reason is that Channel Allocation scheme in works [Alicherry et al., 2005] just considers the number of radios while ignoring the variability of packet arrival rates among different node pairs. This leads to the waste of channel and

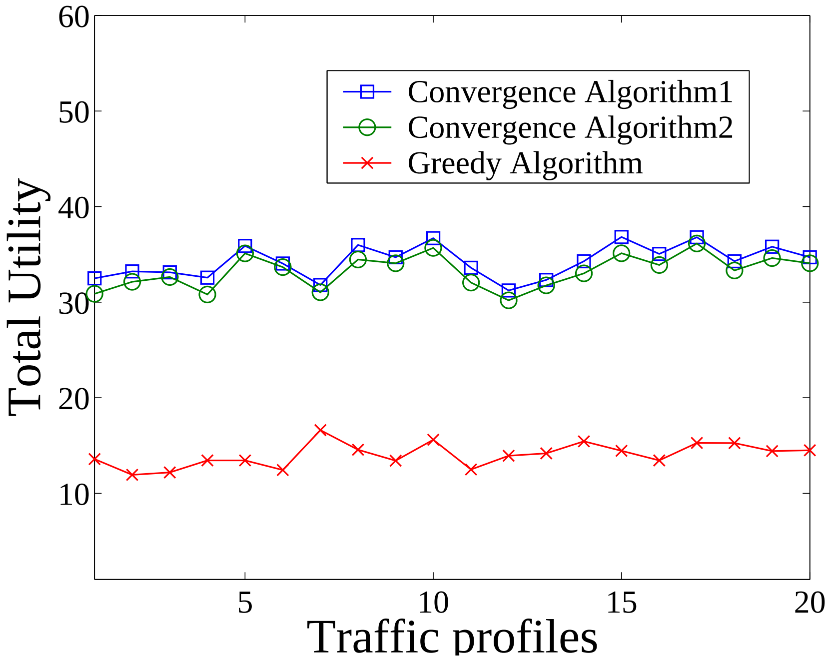


Fig. 5.5: Total utility comparison.

bandwidth resource. The second reason is that it does not consider transmission quality issues. This causes players to overuse bandwidth resource in all channels, which leads to decrease of transmission quality. This explains the reason for the difference between system performance of the stable states converged by our algorithms and NEs in [Alicherry et al., 2005].

5.5 Summary

In this paper, we study the competitive bandwidth and channel allocation problem in multi-radio multi-channel wireless network. Compared with previous works, we consider two new issues, impact of channel traffic load to transmission quality, and variability of packet arrival rates of different players. To solve these two

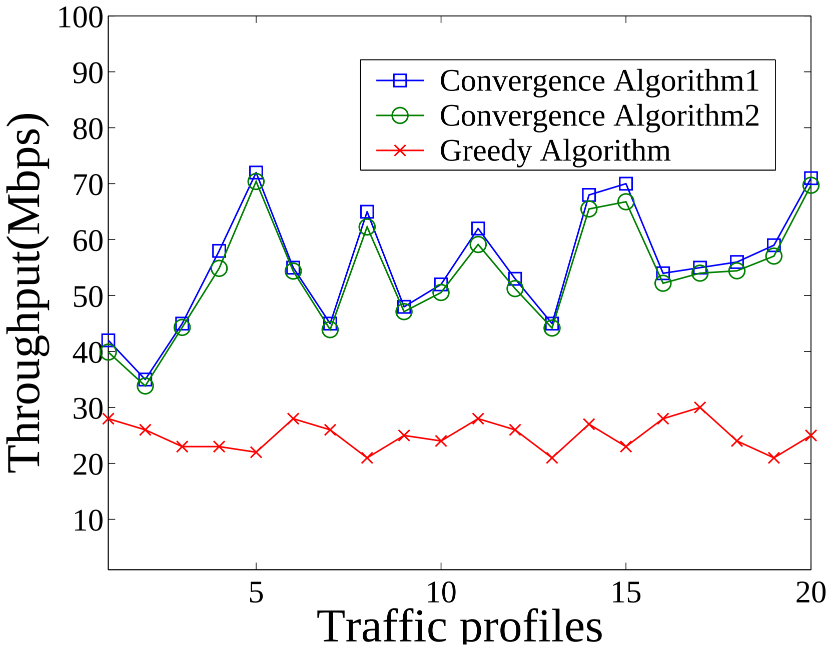


Fig. 5.6: System throughput comparison.

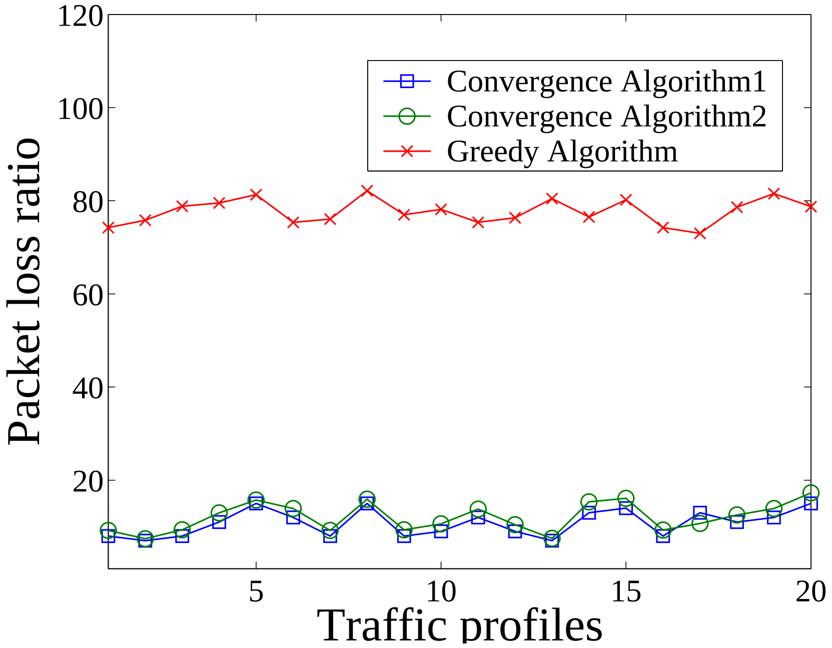


Fig. 5.7: Packet loss ratio comparison.

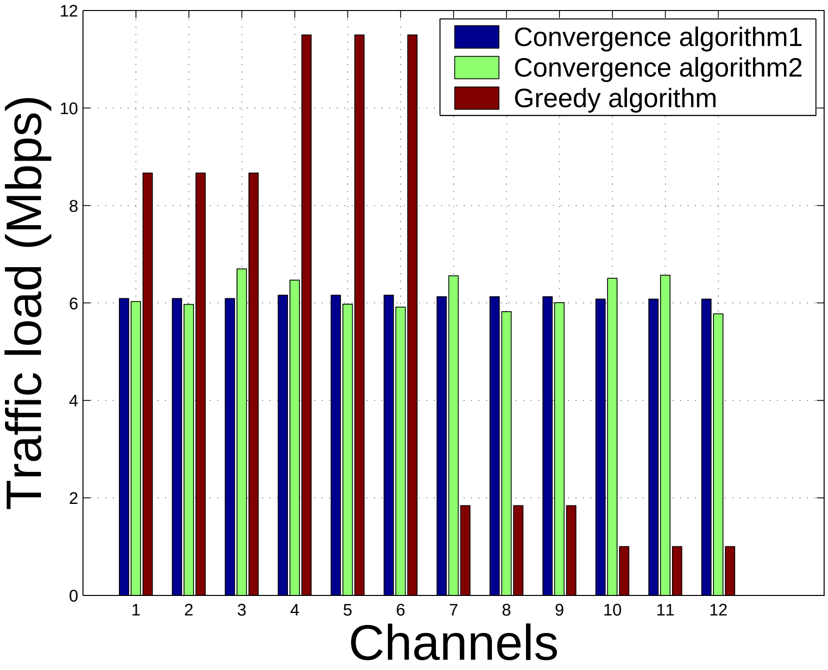


Fig. 5.8: Load balancing property.

new issues, we extend the problem from competitive channel allocation to competitive channel and bandwidth allocation. We model this problem as a non-cooperative static game. We prove the existence of pure NEs for the joint allocation game and the POA of the NEs. We design two distributed algorithms to enable node pairs to converge to NEs. We evaluate the performance of our algorithms by extensive simulations, and find that our algorithms can improve system performance significantly. In terms of future works, we will extend our works of non-cooperative channel and bandwidth allocation to more general network environment, such as multi-hop wireless network, and multi-collision domain networks.

Chapter 6

Conclusions and Future Directions

In this chapter, we summarize main contributions of this dissertation in section 6.1 and point out the future works in 6.2.

6.1 Conclusions

This dissertation investigates the methodology of cross-layer optimization, and utilize this methodology to optimize the performance of wireless networks. We start from analyzing the disadvantages of existing layered network structure. The strict layered network structure limits the flexibility of protocol design, which leads to sub-optimal performance. Thus, the idea of cross-layer design has been introduced as a useful paradigm to improve then network performance. It is especially important to violate the layered network structure in wireless networks because of its limited resource and viable link condition.

In this dissertation, we make three original contributions by applying the cross-layer optimization concept to solve three problems in the wireless network. We propose a unified cross-layer optimization framework to integrate these three works. We describe these three works as follows.

First, we design a cross-layer cooperative QoS routing protocol (CQ-routing) to provide QoS support for multi-hop wireless networks. In this cooperative QoS routing protocol, cooperative communication in the physical layer, scheduling in the MAC layer, and routing in the network layer are jointly considered using cross-layer optimization concept. The cooperative QoS routing protocol includes two parts, the routing algorithm and message mechanism. The routing algorithm targets at finding the routing path with maximum available bandwidth to satisfy users' bandwidth requirement. We propose both centralized and distributed algorithms to find such a routing path. We implement the CQ-routing protocol in our HAWK mesh testbed. We evaluate the performance of the CQ-routing protocol by experiments. The results show that CQ-routing protocol can significantly improve the network performance. Although the routing discovery time becomes longer, we argue that it is tolerable since one flow only needs to find a cooperative routing path once.

Second, we study multi-link spectrum handoff problem to maintain the network connectivity and minimize the Total Handoff Completion Time (THCT). In this spectrum handoff coordination protocol, we jointly consider spectrum handoff issue in the physical layer and rerouting issue in the network layer. Existing work only considered the problem of minimizing spectrum handoff delay of a single link in single-hop cognitive networks, referred to as the SH-SLSH problem. This paper studies a more challenging problem (referred to as the SH-MLMH problem) in which multiple links perform spectrum handoff in multi-radio multi-hop cognitive networks. In multi-hop cognitive networks, multi-link spectrum handoff can significantly influence the performance of the cognitive network. Using a cross-layer approach, we propose both centralized and distributed SHS algorithms to solve the $SH-MLMH$ problem. We prove that the centralized SHS algorithm can achieve a logarithmic approximation ratio, and the distributed SHS algorithm can generate valid solutions. Simulation results show that the network performance can be significantly improved by SHS algorithms. The results of this paper can be applied in the future cognitive network, and promote the cognitive radio to real application.

Third, we study the competitive bandwidth and channel allocation problem in multi-radio multi-channel wireless network. In this problem, we jointly consider the channel allocation issue in the physical layer and the bandwidth allocation issue in the MAC layer. Compared with previous works about competitive channel allocation, we consider two new issues, impact of channel traffic load to transmission quality, and variability of packet arrival rates of different players. To solve these two new issues, we extend the problem from competitive channel allocation to competitive channel and bandwidth allocation. We model this problem as a non-cooperative static game. We prove the existence of pure NEs for the joint allocation game and the POA of the NEs. We design two distributed algorithms to enable node pairs to converge to NEs. We evaluate the performance of our algorithms by extensive simulations, and find that our algorithms can improve system performance significantly. In terms of future works, we will extend our works of non-cooperative channel and bandwidth allocation to more general network environment, such as multi-hop wireless network, and multi-collision domain networks.

In summary, using the cross-layer optimization methodology, we propose three protocols, which target at satisfying users' QoS requirement, minimize total spectrum handoff delay, and maximize throughput. The results show that the cross-layer optimization paradigm can perform better the layered one. Therefore, a new wireless network which violate the layered structure will be demanded in the future.

6.2 Future Works

We close this dissertation by pointing out future works. Specifically, we plan to do three major works in the future.

First, we plan to study the problem of how to maximize the cooperative multi-hop wireless networks. In our work about CQ-routing protocol, we only study how to utilize cooperative communication to maximize available bandwidth for single flow. In the future, we plan to utilize cooperative communication to improve the network throughput of a multi-hop wireless network. In the new problem, we need to consider interference relationship among multiple flows. Specifically speaking, we will study throughput maximization problem for multi-hop cooperative wireless networks - given a set V of wireless nodes, and a set D of connections, assuming that a Decode-and-Forward (DF) cooperative scheme is adopted, what is the maximum total transmission rates achieved by these connections? How to design protocols to achieve the maximum throughput? We need to address new challenging issues, such as the relay node selection, and more complicated interference relationship. We believe that maximizing throughput of cooperative multi-hop wireless network will promote the applications of cooperative communication in the real wireless network.

Second, we plan to study the problem of how to design cooperative QoS routing protocol in the multi-radio multi-channel wireless networks. In our previous works about cooperative QoS routing, we only study cooperative routing issue in the single-radio single-channel wireless network. In the future, we try to integrate two new physical technologies, cooperative communication and multi-radio multi-channel technology. We can find that the result of channel allocation for utilizing multi-radio multi-channel technology will impact the selection of relay nodes for cooperative communication. Therefore, channel allocation and relay node selection should be jointly considered. Moreover, to provide QoS support in multi-hop wireless network, we also need to jointly consider two other issues, including scheduling in the MAC layer and routing in the network layer. This will significantly complicate the problem, and require us to design new protocols to solve this problem.

Finally, we plan to study the non-cooperative channel and bandwidth allocation problem in the multi-hop wireless network. In our previous works about cooperative QoS routing, we only study cooperative routing issue in the single-radio single-channel wireless network. In the future, we need to jointly study another new issue, routing together with channel and bandwidth allocation. We will also use game theory to analyze the problem. We will model this problem as a game, and prove if there exists pure Nash Equilibrium for the problem. We will also design convergence algorithm to enable nodes to converge to the stable state. The unique challenge for the multi-hop wireless network is the complicated topology relationship and interference relationship. It will be more challenging for us to analyze the stable state and design convergence algorithm. However, we believe that it is meaningful to study the problem in the multi-hop wireless network, which will promote the application of the multi-radio multi-channel technology in the multi-hop wireless network.

Bibliography

Aghvami, A. and T.Le, N. O. (2001). Mode switching and qos issues in software radio. *IEEE Personal Communications*, 5:38–44.

Agrawal, P., Chen, S., Ramanathan, P., and Sivalingam, K. (1998). Battery power sensitive video processing in wireless networks. In *proceedings of IEEE PIMRC 1998, Boston, USA*.

Akyildiz, I. F., Lee, W. Y., Vuran, M., and Mohanty, S. (2006). Next generation dynamic spectrum access cognitive radio wireless networks: A survey. *Computer Networks Journal (Elsevier)*, 50:2127–2159.

Alicherry, M., Bhatia, R., and Li, L. (2005). Joint channel assignment and routing for throughput optimization in multi-radio wireless mesh networks. In *proceedings of ACM MobiCom 2005, Cologne, Germany*.

Alwan, A., Bagrodia, R., Bambos, N., Gerla, M., Kleinrock, L., Short, J., and Villasenor, J. (2002). Adaptive mobile multimedia networks. *IEEE Personal Communications*, 3:34–51.

B.Davie, A.Charny, and K.Benson (2002). *RFC3246: An Expedited Forwarding PHB (Per-Hop Behavior)*. ITF,Network Working Group.

Bertsekas, D. and Gallager, R. (1992). *Data Networks*. Prentice Hall.

Bianchi, G. (2000). Performance analysis of the ieee 802.11 distributed coordination strategy. *IEEE Journal on Selected Areas in Communication*, 18:1927–1948.

Bondy, J., Adrian, and Murty (2006). *Graph Theory*. Springer.

Caceres, R. and Iftode, L. (2002). Improving the performance of reliable transport protocols in mobile computing environments. *IEEE Journal on Selected Areas in Communications,*, 13:850–857.

Cao, J., Xie, K., Wu, W., and etc (2009). Hawk: Real-world implementation of high-performance heterogeneous wireless network for internet access. In *proceedings of the 1st ICDCS International Workshop on Next Generation Network Architectures (NGNA'09), Montreal, Quebec, Canada.*

Chafekar, D., Kumar, V. S. A., Marathe, M. V., Parthasarathy, S., and Srinivasan, A. (2008). Approximation algorithms for computing capacity of wireless networks with sinr constraints. In *proceedings of IEEE INFOCOM 2008, Phoenix, USA.*

Chazelle (2000). A minimum spanning tree algorithm with inverse-ackermann type complexity. *Journal of ACM*, 47:1028–1047.

Chen, F., Zhai, H., and Fang, Y. (2010). Available bandwidth in multirate and multihop wireless ad hoc networks. *IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS*, 28:299–307.

Chen, Q., Zhang, Q., and Niu, Z. (2008). Qos-aware cooperative and opportunistic scheduling exploiting multiuser diversity for rate-adaptive ad hoc networks. *IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY*, 57(3):1113–1125. 3830.

Chia-Cheng Hu, Yu-Liang Kuo, C.-Y. C. Y.-M. H. (2010). Maximum bandwidth routing and maximum flow routing in wireless mesh networks. *Telecommunication Systems doi:10.1007/s11235-009-9217-2*, 10(3):125–134. 3830.

Cordeiro, C., Challapali, K., and Birru, D. (2006). IEEE 802.22: An introduction to the first wireless standard based on cognitive radios. *Journal of Communications*, 1:1927–1948.

Das, A. K., Alazemi, H. M., Vijayakumar, R., and Roy, S. (2005). Optimization models of fixed channel assignment in wireless mesh networks with multiple radios. In *proceedings of IEEE SECON 2005, Santa Clara, CA, USA*.

D.Cohen (2005). *Spectrum Regulations for Ad Hoc Wireless Cognitive Radios*. FCC.

DeSimone, A., M.Chuah, and Yue, O. (2003). Throughput performance of tranport-layer protocols over wireless lands. In *proceedings of IEEE GLOBECOM 2003, San Francisco, USA*.

D.Malone, K. and D.Leith (2007). Modeling the 802.11 distributed coordination strategy in nonsaturated heterogeneous conditions. *IEEE TRANSACTIONS ON NETWORKING*, 15:159–172.

DU, K. and YANG, Y. (2010). A qos routing for maximum bandwidth in ad hoc networks. In *the Second International Conference on Future Networks*, volume 2, pages 1012–1017.

Ebert, J. and Wolisz, A. (1999). Combined tuning of rf power and medium access control for wlans. In *proceedings of 1999 IEEE International Workshop on Mobile Multimedia Communications (MoMuC'99), San Diego, USA*.

Felegyhazi, M., Cagalj, M., Bidokhti, S., and Hubaux, J. (2007). Non-cooperative multi-radio channel allocation in wireless networks. In *proceedings of IEEE INFOCOM 2007, Anchorage, Alaska, USA*.

Fudenberg, D. and Tirole, J. (1991). *Game Theory*. Cambridge, MA: MIT Press.

Gao, L. and Wang, X. (2008). A game approach for multi-channel allocation in multi-hop wireless networks. In *proceedings of the 9th ACM international symposium on Mobile ad hoc networking and computing (Mobihoc 2008), Hong Kong, China*.

G.Gottlob, G.Greco, and F.Scarcello (2005). Pure nash equilibria: Hard and easy games. *Journal of Artificial Intelligence Research*, 24:357–406.

Ghasemi, A. and Sousa, E. (2007). Optimization of spectrum sensing for opportunistic spectrum access in cognitive radio networks. In *proceedings of IEEE Consumer Communications and Networking Conference 2007 (CCNC'07), Harrah, Las Vegas, Nevada, USA*.

Gupta, D. and Wu, D. (2009). Experimental comparison of bandwidth estimation tools for wireless mesh networks. In *proceedings of IEEE Infocom 2009, Rio de Janeiro, Brazil*.

Gurashish Brar, Douglas M. Blough, P. S. (2006). Computationally efficient scheduling with the physical interference model for throughput improvement in wireless mesh networks. In *MobiCom06, Los Angeles, California, USA.*, volume 2, pages 1012–1017.

H.Kim and K.G.Shin (2008). Fast discovery of spectrum opportunities in cognitive radio networks. In *proceedings of IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks 2008 (DYSPAN'08), Chicago, Illinois, USA.*

Ibrahim, A. S., Han, Z., and Liu, K. J. R. (2008). Distributed energy-efficient cooperative routing in wireless networks. *IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS*, 8(3):3930–3941. 3830.

Jia, J., Zhang, Q., and Shen, X. (2008). Hc-mac: a hardware-constrained cognitive mac for efficient spectrum management. *IEEE Journal on Selected Areas in Communications*, 26:106–117.

J.S.Wu, C.Cheng, N.Huang, and G.Ma (2001). Intelligent handoff for mobile wireless internet. *IEEE Special Issue on Wireless Internet and Intranet Access*, 6:417–426.

Jung, E. and Liu, X. (2008). Opportunistic spectrum access in heterogeneous user environments. In *proceedings of IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks 2008 (DYSPAN'08), Chicago, Illinois, USA.*

Khalife, H., Ahuja, S., Malouch, N., and Krunz, M. (2008). Probabilistic path selection in opportunistic cognitive radio networks. In *proceedings of IEEE GLOBECOM 2008, New Orleans, LA, USA.*

Kim, H. and Shin, K. (2008). Efficient discovery of spectrum opportunities with mac-layer sensing in cognitive radio networks. *IEEE Transactions on Communication*, 7:533–545.

Kodialam, M. and Nandagopal, T. (2005). Characterizing the capacity region in multi-radio multi-channel wireless mesh networks. In *ACM MobiCom 2005, Cologne, Germany.*

Kravets, R. and Krishnan, P. (2000). Application-driven power management for mobile communication. *IEEE Wireless Networks,*, 6:263–277.

Kuhn, F., Wattenhofer, R., and Zollinger, A. (2003). Worstcase optimal and averagecase efficient geometric adhoc routing. In *proceedings of ACM MobiHoc 2003, Annapolis, Maryland, USA.*

Kyasanur, P. and Vaidya, N. H. (2005). Algorithm design challenges for multi-hop dynamic spectrum access networks. In *proceedings of IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks 2005 (DYSPAN'05), Baltimore, Maryland, USA.*

Laneman, J. N., Tse, D. N. C., and Wornell, G. W. (2004). Cooperative diversity in wireless networks: Efficient protocols and outage behavior. *IEEE TRANSACTIONS ON INFORMATION THEORY*, 50:3062–3080.

Lee, W. Y. and Akyildiz, I. (2008). Optimal spectrum sensing framework for cognitive radio networks. *IEEE Transactions on Wireless Communications*, 7:3845–3857.

Lei, G., W, W., Peng, T., and Wang, W. (2008). Routing metrics in cognitive radio networks. In *proceedings of the 4th IEEE International Conference on In Circuits and Systems for Communications (ICCSC'08), Shanghai, China.*

Lettieri, P. and Srivastava, M. (1998). Adaptive frame length control for improving wireless link throughout,range and energy efficiency. In *proceedings of IEEE INFOCOM 1998, San Francisco, CA, USA.*

Li, F., Wu, K., and Lippman, A. (2006). Energy-efficient cooperative routing in multi-hop wireless ad hoc networks. In *IEEE International Performance, Computing, and Communications Conference*, volume 2, pages 1012–1017.

Lin, Y., H.Song, J., and Wong, V. W. S. (2008). Cooperative protocols design for wireless ad-hoc networks with multi-hop routing. *IEEE Mobile Networks and Applications*, 14:143–153.

Liu, H. and Zarki, M. (2003). Adaptive source rate control for real-time wireless video transmission. *Mobile Networks and Applications*, 3:49–60.

Liu, P., Z.F.Tao, Narayanan, S., Korakis, T., and Panwar, S. S. (2007). Coopmac: A cooperative mac for wireless lans. *IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS*, 25:340–354.

L.Luo and S.Roy (2007). Analysis of search schemes in cognitive radio. In *proceedings of IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks 2007 (WOCN'07), Salt Lake City, Utah, USA*.

Ludwig, R., Konrad, A., Joseph, A., and Katz, R. (2002). Optimizing the end-to-end performance of reliable flows over wireless links. *IEEE Wireless Networks,*, 8:289–299.

Luo, J., Blum, R. S., Greenstein, L. J., Cimini, L. J., and M.Haimovich, A. (2006a). New approaches for cooperative use of multiple antennas in ad hoc wireless networks. In *IEEE 60th Vehicular Technology Conference*, volume 2, pages 1012–1017.

Luo, L., Gruteser, M., Liu, H., Raychaudhuri, D., Huang, K., and Chen, S. (2006b). A qos routing and admission control scheme for 802.11 ad hoc networks. In *proceedings of ACM DIWANS 2006, Los Angeles, California, USA*.

Ma, H., Zheng, L., Ma, X., and Luo, Y. (2008). Spectrum-aware routing for multi-hop cognitive radio networks with a single transceiver. In *proceedings Cognitive Radio Oriented Wireless Networks and Communications 2008 (CrownCom'08), Singapore*.

Maham, B., Debbah, M., and Hjrungnes, A. (2008). Energy-efficient cooperative routing in ber constrained multihop networks. In *proceedings of the 3rd international conference on Communications and Networking in China (ChinaCom'08), Hangzhou, China*.

Methfessel, M. and Dombrowski, K. (2002). Vertical optimization of data transmission for mobile wireless terminals. *IEEE Wireless Communications*, 6:36–43.

Mishra, A., Banerjee, S., and Arbaugh, W. (2005). Weighted coloring based channel assignment for wlans. *ACM SIGMOBILE Mobile Computing and Communications Review (MC2R)*, 9:19 – 31.

M.R.Garey and D.S.Johnson (2005). *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Freeman Company.

Mukherjee, A. and Kwon, H. M. (2010). General auction-theoretic strategies for distributed partner selection in cooperative wireless networks. *IEEE TRANSACTIONS ON COMMUNICATIONS*, 58(3):2903–2915. 3830.

Narayanan, L. and Shende, S. (2001). Static frequency assignment in cellular networks. *Algorithmica*, 29:396–409.

Nisan, N., Roughgarden, T., Tardos, E., and Vazirani, V. (2005). *Algorithmic Game Theory*. Cambridge Press.

Noble, B., Satyanarayanan, M., Narayanan, D., Tilton, J., Flinn, J., and Walker, K. (1997). Agile application-aware adaptation for mobility. In *proceedings of the 16th ACM Symposium on Operating System Principles (SOSP-16), Saint-Malo, France.*

Nosratinia, A., Hunter, T., and Hedayat, A. (2004). Cooperative communication in wireless networks. *IEEE Communications Magazine*, 42:68–73.

Novak, L., Karadzic, Z., and Acketa, D. (2005). *A Space Optimal Algorithm for Enumerating Spanning Trees of a Connected Graph*. Springer.

Pandana, C., Siriwongpairat, W. P., Himsoon, T., and Liu, K. J. R. (2006). Distributed cooperative routing algorithms for maximizing network lifetime. In *IEEE Wireless Communications and Networking Conference (WCNC)*, volume 2, pages 1012–1017.

Pei, Y., Hoang, A., and Liang, Y. C. (2007). Sensing throughput tradeoff in cognitive radio networks: how frequently should spectrum sensing be carried out? In *proceedings of IEEE International Symposium on Personal, Indoor and Mobile Radio Communications 2007(PIMRC'07), Athens, Greece.*

Phan, K. T., Nguyen, D. H. N., and Le-Ngoc, T. (2010). Joint power allocation and relay selection for multiuser cooperative communication. *IEEE Transactions on Wireless Communications*, 9(3):1255–1260. 3830.

Raisinghani, V. and Iyer, S. (2003). User managed wireless protocol stacks. In *proceedings of IEEE ICDCS 2003, Rhode Island, USA.*

Raisinghani, V., Singh, A., and Iyer, S. (2002). Improving tcp performance over mobile wireless environments using cross-layer feedback. In *proceedings of IEEE ICPWC 2002, New Delhi, India.*

Raniwala, A., Gopalan, K., and cker Chiueh, T. (2004a). Centralized channel assignment and routing algorithms for multi-channel wi reless mesh networks. *ACM Mobile Computing and Communications Review*, 8(3):50–65. 3830.

Raniwala, A., Gopalan, K., and T.Chiueh (2004b). Centralized channel assignment and routing algorithms for multi-channel wireless mesh networks. *ACM SIGMOBILE Mobile Computing and Communications Review (MC2R)*, 8:50–65.

Raniwala, A. and T.Chiueh (2005). Architecture and algorithms for an ieee 802.11-based multi-channel wireless mesh network. In *proceedings of IEEE INFOCOM 2005, Miami, FL, USA.*

Sanmateu, A., F. Paint, L. M., Tessier, S., Fouquart, P., Sollund, A., and Bustos, E. (2002). Seamless mobility across ip networks using mobile ip. *IEEE Computer Networks*, 40:181–190.

Sendonaris, A., Erkip, E., and Aazhang, B. (2003). User cooperation diversity part i and part ii. *IEEE TRANSACTIONS ON COMMUNICATIONS*, 51:1927–1948.

Sharma, S., Shi.Y, Y., Hou, T., Sherali, H. D., and Kompella, S. (2010). Cooperative communications in multi-hop wireless networks: Joint flow routing and relay node assignment. In *proceedings of IEEE INFOCOM 2010, San Diego, CA, USA.*

Shi, Y., Sharma, S., Hou, Y. T., and Kompella, S. (2008). Optimal relay assignment for cooperative communications. In *proceedings of ACM MobiHoc 2008, Hong Kong, China.*

Shiang, H. P. and Schaar, M. (2008). Delay-sensitive resource management in multihop cognitive radio networks. In *proceedings of IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks 2008 (DYSPAN'08), Chicago, Illinois, USA.*

Shin, M., Lee, S., and Kim, Y. (2006). Distributed channel assignment for multi-radio wireless networks. In *proceedings of IEEE Mobile Adhoc and Sensor Systems 2006 (MASS'06), Vancouver, Canada.*

Sikora, M., Laneman, J. N., Haenggi, M., Costello, D. J., and Fuja, T. E. (2006). Bandwidth- and power-efficient routing in linear wireless networks. *IEEE Trans. Inform. Theory*, 8(3):2624–2633. 3830.

Tang, J., Xue, G., and Zhang, W. (2008). Interference-aware topology control and qos routing in multi-channel wireless mesh networks. In *proceedings of ACM MobiHoc 2008, Hong Kong, China.*

Tian, J. and Bi, G. (2006). A new link maintenance and compensation model for cognitive uwb radio systems. In *proceedings of International Conference on ITS Telecommunications 2006 (ITST'06), Chengdu, China.*

Tong, L., Naware, V., and Venkitasubramaniam, P. (2004). Signal processing in random access. *IEEE Signal Processing Magazine*, 21:29–39.

van den Heuvel, J., Leese, R. A., and Shepherd, M. A. (1998). Graph labeling and radio channel assignment. *Journal of Graph Theory*, 29:263–283.

Wang, L. C. and Chen, A. (2008). On the performance of spectrum handoff for link maintenance in cognitive radio. In *proceedings of the 3rd International Symposium on Wireless Pervasive Computing (ISWPC 2008), Santorini, Greece.*

Wang, P., Xiao, L., Zhou, S., and Wang, J. (2007). Optimization of detection time for channel efficiency in cognitive radio systems. In *proceedings of IEEE Wireless Communications and Networking Conference 2007 (WCNC'07), HongKong, China.*

Wang, Q. and Zheng, H. (2006). Route and spectrum selection in dynamic spectrum networks. In *proceedings of IEEE Consumer Communications and Networking Conference 2006 (CCNC'06), Harrah, Las Vegas, Nevada, USA.*

Willkomm, D., Gross, J., and Wolisz, A. (2005). Reliable link maintenance in cognitive radio systems. In *proceedings of IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks 2005(DYSPAN'05), Baltimore, Maryland, USA.*

Wu, F., Zhong, S., and Qiao, C. (2008). Globally optimal channel assignment for non-cooperative wireless networks. In *proceedings of IEEE INFOCOM 2008, Phoenix, AZ, USA.*

Wu, J. and Li, H. (1999). On calculating connected dominating set for efficient routing in ad hoc wireless networks. In *proceedings of ACM DlALM 1999, Scattlc, WA, USA.*

Xin, C. (2005). A novel layered graph model for topology formation and routing in dynamic spectrum access networks. In *proceedings of IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks 2005 (DYSPAN'05), Baltimore, Maryland, USA*.

Xin, C., Ma, L., and Shen, C.-C. (2008). A path-centric channel assignment framework for cognitive radio wireless networks. *Mobile Networks and Applications*, 13:463–476.

Xylomenos, G. and Polyzos, G. (2001). Quality of service support over multiservice wireless internet links. *IEEE Computer Networks*, 5:601–615.

Yang, L., Cao, L., and Zheng, H. (2007). Proactive channel access in dynamic spectrum network. In *the Second IEEE International Conference on Cognitive Radio Oriented Wireless Networks and Communications 2007 (CrownCom'07),*.

Yu and Richard, F. (2011). *CRAHNs: Cognitive radio ad hoc networks*. springer.

Zhai, H. and Fang, Y. (2006). Impact of routing metrics on path capacity in multirate and multi-hop wireless ad hoc networks. In *proceedings of the 14th IEEE International Conference on Network Protocols 2006 (ICNP06), Santa Barbara, California, USA*.

Zhang, J., Jia, J., Zhang, Q., and Lo, E. M. K. (2010). Implementation and evaluation of cooperative communication schemes in software-defined radio testbed. In *proceedings of IEEE INFOCOM 2010, San Diego, CA, USA*.

Zhang, J. and Zhang, Q. (2008). Cooperative routing in multi-source multi-destination multi-hop wireless networks. In *proceedings of IEEE INFOCOM 2008, Phoenix, USA*.

Zheng, H. and L.Cao (2005). Device-centric channel management. In *proceedings of IEEE DYSPAN 2005, Baltimore, Maryland, USA*.

Zhu, C. and M.Corson (2002). Qos routing for mobile ad hoc networks. In *proceedings of IEEE Infocom 2002, New York, USA*.

Zhu, G., Felice, M., and Akyildiz, I. (2008). Stod-rp: a spectrum-tree based ondemand routing algorithm for multi-hop cognitive radio networks. In *proceedings of IEEE GLOBECOM 2008, New Orleans, LA, USA*.

Zhu, X., Shen, L., and Yum, T. S. P. (2007). Analysis of cognitive radio spectrum access with optimal channel reservation. *IEEE Communications Letters*, 11:304–306.