Improving the Capacity of Wireless Ad Hoc Networks through Multiple Channel Operation: Design Principles and Protocols

Michelle Xiaohong Gong

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Electrical Engineering

Scott F. Midkiff, Chair
R. Michael Buehrer
Ing-Ray Chen
Luiz A. DaSliva
C. Patrick Koelling

June 7, 2005
Blacksburg, Virginia

Keywords: Ad hoc routing protocol, cross-layer design, distributed channel assignment, multi-channel medium access control, wireless ad hoc networks

Improving the Capacity in Wireless Ad Hoc Networks through Multiple Channel Operation: Design Principles and Protocols

Michelle Gong

(ABSTRACT)

Despite recent advances in wireless local area network (WLAN) technologies, today's WLANs still cannot offer the same data rates as their wired counterparts. The throughput problem is further aggravated in multi-hop wireless environments due to collisions and interference caused by multi-hop routing. Because all current IEEE 802.11 physical (PHY) standards divide the available frequency into several orthogonal channels, which can be used simultaneously within a neighborhood, increasing capacity by exploiting multiple channels becomes particularly appealing.

To improve the capacity of wireless ad hoc networks by exploiting multiple available channels, I propose three principles that facilitate the design of efficient distributed channel assignment protocols. Distributed channel assignment problems have been proven to be $\mathcal{NP}$-complete and, thus, computationally intractable [32, 6]. Though being a subject of many years of research, distributed channel assignment remains a challenging problem. There exist only a few heuristic solutions, none of which is efficient, especially for the mobile ad hoc environment [6, 23, 32]. However, protocols that implement the proposed design principles are shown to require fewer channels and exhibit significantly lower communication, computation, and storage complexity, compared with existing approaches. As examples, I present two such protocols that build on standard reactive and proactive routing protocols. In addition, I prove the correctness of the algorithms and derive an upper bound on the number of channels required to both resolve collisions and mitigate interference.

A new multi-channel medium access control (MC-MAC) protocol is also proposed for multi-hop wireless ad hoc networks. MC-MAC is compatible with the IEEE 802.11 medium access control (MAC) standard and imposes the minimum system requirements among all existing multi-channel MAC protocols. In addition, simulation results show that even with only a single half-duplex transceiver, MC-MAC, by exploiting multiple channels, can offer up to a factor of four improvement in throughput over the IEEE 802.11 MAC protocol. The reduction in delay is even more significant. Therefore, the MC-MAC protocol and the accompanying distributed channel assignment protocols constitute an effective solution to the aforementioned performance problem in a multi-hop wireless network.

Finally, I generalize the cross-layer design principle to more general networking functions and present a network architecture to motivate and facilitate cross-layer designs in wireless networks. A literature survey is provided to validate the proposed cross-layer design architecture. Current cross-layer design research can be categorized into two classes: joint-layer design using optimization techniques, and adaptive techniques based on system-profile and/or QoS requirements. Joint-layer design based on optimization techniques can achieve optimal performance, but at the expense of complexity. Adaptive schemes may achieve relatively good performance with less complexity. Nevertheless, without careful design and a holistic view of the network architecture, adaptive schemes may actually cause more damage than benefit.

Acknowledgments

When I arrived at Virginia Tech three years ago for my Ph.D. degree, I felt being a "tiny and hideous caterpillar," sealed in a cocoon, but dreaming to get out one day and to be able to fly just like all beautiful butterflies. During my whole first year at Blacksburg, I stayed in the cocoon and gathered up my strength. Little by little, I felt I had accumulated enough strength and started to look for a way out. Unfortunately, everywhere I turned, I faced obstacles. It seemed that there was no way for me to get out, and my butterfly dream seemed un-accomplishable. "Perhaps, I am not a butterfly material indeed." I thought to myself. But suddenly, I heard a voice coming to me from outside of the cocoon. Oh, I realized I was not alone at all.

In fact, every caterpillar has a primary helper, called an "advisor". My advisor is Dr. Scott F. Midkiff, who has offered so much great help to me. As an experienced caterpillar helper, Dr. Midkiff understands that the struggling process getting out of the cocoon is the single most important step towards becoming a butterfly. Even though he has "a pair of scissors" in his hands, he did not just cut the cocoon open and release me out. He knows, by doing so, I would never be able to fly [1].

Knowing that there is no easy way out, I had to struggle by myself to get out. Dr. Midkiff watched me as I made a small opening on the cocoon. "Good direction!" he instructed. Feeling encouraged, I started to work harder and made the opening bigger. Getting out of the cocoon was a painful and exhausting process. Fortunately, Dr. Midkiff was always watching me and at my side. He inspired me when I got stuck, encouraged me when my spirit was down, and challenged me when I was feeling sluggish. If I can ever become a butterfly flying strong into the sky, I will be most grateful to my advisor, Dr. Midkiff.

Besides my primary helper, I am fortunate enough to have many other crucial helpers as well. Dr. R. Michael Buehrer's in-depth knowledge in communication theory and practices has enlightened me and gave me an early start on my research. Dr. Ing-Ray Chen's enthusiasm on research and his style on conducting research to achieve a high precision have always inspired me. I like to be challenged by Dr. Chen because I know that challenges will make me a stronger butterfly. Dr. Luiz A. DaSilva is always there when I have a technical question. Even though he has many "caterpillars" of his own to take care, he is so down-to-earth and so patient to everyone who asks him questions. Dr. DaSilva also encouraged me to pay attention to details, one of the most important qualities of good researchers. Last but not the least, Dr. Patrick Koelling taught me how to become a better researcher. "Everyone knows how to do simulations, but not everyone can do it right." he said in the simulation class. Not only did I learn quite a few simulation and optimization techniques from him, I also gained important research skills. Thanks to Dr. Koelling, I will always remember to define long enough warm-up periods in my simulations.

I am also very grateful to Dr. Shiwen Mao, Dr. Peng Li, and Dr. Tao Lin. All three of them are my predecessors and were once "caterpillars" themselves. They have been through the process and have become strong "butterflies". They shared their experiences with me, encouraged me, and taught me many techniques to get out of the cocoon. I would not have made it without them.

Then, there are all these "cheerleaders" who have always been with me during my lonely struggle process – my hiking and skiing buddy, Otis Farmer; my lunch partners, Unghee Lee and Rutger Thomschutz; the "Big Teddy Bear of the lab," George Hadjichristofi; generous and warm-hearted Mike Thompson who has helped most of us move at least one piece of furniture; and the quiet Kerry Wood who always has a nice smile on his face.

I am forever thankful to my parents, Zhanping Gong and Shouru Zhang, who not only brought me to this world, but also taught me to be self-dependent and resilient, my most important assets.

I also wish to acknowledge the National Science Foundation, which supported me during my Ph.D. studies through an Integrated Graduate Education and Research Training (IGERT) grant, award DGE-9987586.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

Despite recent advances in wireless local area network (WLAN) technologies, today's WLANs still cannot offer the same data rates as their wired counterparts. The throughput problem is further aggravated in multi-hop wireless environments due to *intra-flow interference* introduced by adjacent nodes on the same path and *inter-flow interference* generated by nodes from neighboring paths. For instance, it has been shown by Li, *et al.* [47] that the maximum capacity that the IEEE 802.11 medium access control (MAC) protocol can achieve for a chained network is just one seventh of the available bandwidth.

All current IEEE 802.11 physical (PHY) standards divide the available frequency into several orthogonal channels, which can be used simultaneously within a neighborhood. Therefore, increasing capacity by exploiting multiple channels becomes particularly appealing. In fact, such bandwidth aggregation has been widely used in infrastructure-based WLANs, where high-end access points have multiple interfaces that operate on different channels simultaneously [59]. In infrastructure-based WLANs, non-overlapping channels are distributed among different access points at the network planning stage. However, IEEE 802.11 WLANs that operate in ad hoc mode rarely use multiple channels simultaneously partly because the IEEE 802.11 MAC is not designed to operate with multiple channels. Therefore, an ad hoc network that is based on IEEE 802.11a technology utilizes only one out of 12 available orthogonal channels, wasting more than 90 percent of the potentially available spectrum. If multiple networks co-locate in a close proximity, each of them may utilize a different channel. Nevertheless, my research goal is to improve performance for a single network through utilizing multiple channels.

Consequently, there has been substantial interest in multi-channel MAC schemes that can achieve higher throughput by exploiting multiple available channels [76, 35, 59]. Some early investigations, e.g. [67] and [36], assumed that every node has its own unique channel. Therefore, no channel assignment or selection is needed. However, in reality, the number of channels is limited and has to be carefully assigned to each node to avoid contention and collisions and to enable optimal spatial re-use of available channels.

The need for spatial reuse of available channels motivates research on distributed channel assignment. Distributed channel assignment protocols assign distinct channels to nodes in a network to allow multiple transmissions to occur simultaneously without incurring collisions that degrade performance. However, distributed channel assignment problems have been proven to be $\mathcal{NP}$-complete and, thus, computationally intractable [32, 6]. Despite being the subject of many years of research, distributed channel assignment remains a challenging problem. There exist only a few heuristic solutions, none of which is efficient, especially for the mobile ad hoc environment [6, 23, 32]. Some solutions have such high complexity that even simulation studies become difficult [23]. In addition, existing algorithms assume that the number of channels is sufficiently large to always avoid interference. Since they do not take into consideration the case when there are not enough channels, convergence, i.e. reaching a desired stable result, may become a problem.

Furthermore, most of distributed channel assignment protocols only consider secondary collisions, since they are mainly designed to solve the “hidden terminal” problem. Nevertheless, primary collisions and interference are also important factors that adversely affect channel utilization and network capacity. *Primary collisions* occur when two neighboring nodes transmit to each other at the same time, whereas *secondary collisions* occur when transmitters outside of radio range of each other, also called “hidden terminals,” transmit to the same receiver [6]. Primary collisions can be avoided or reduced by using random access protocols, such as ALOHA [3] or CSMA [42], while secondary collisions can be avoided by multi-channel medium access schemes or handshake mechanisms, such as request-to-send/clear-to-send (RTS/CTS) signalling [2]. However, interference generated by nodes that are two or more hops away cannot be avoided by random access schemes or handshake mechanisms and, thus, such interference is potentially more harmful. Failing to take primary collisions and interference into consideration, existing channel assignment algorithms may suffer performance degradation, especially in a multi-hop environment.

I propose three principles for efficient distributed channel assignment design [28]. First, to reduce the complexity of the channel assignment algorithm, channel assignment and routing are combined together. This “cross-layer” design principle is motivated by the fact that both the channel assignment algorithm and the ad hoc routing algorithm must be invoked when there is a change in the network topology. Utilizing this design principle can greatly reduce the complexity of channel assignment algorithms.

Secondly, channels should be assigned only to “active” nodes. This “on-demand” channel assignment principle is motivated by the fact that only active nodes need valid channels. Existing channel assignment schemes assign channels to all nodes in the network regardless of whether they are active or not. However, if this on-demand assignment principle is implemented, fewer channels may be required in the network.

Finally, both primary and secondary collisions and interference should be taken into consideration. To mitigate interference as well as to resolve collisions, distinct channels should be assigned in a way that collisions and interference can be avoided as much as possible.

As examples, I present two new channel assignment protocols that implement these design principles. In general, wireless ad hoc routing protocols can be classified into two categories: *reactive* and *proactive*. Three Internet Engineering Task Force (IETF) Request For Comments (RFCs) on wireless ad hoc routing have been published. RFC 3661 describes a reactive routing protocol, called Ad hoc On-demand Distance Vector (AODV) routing [55]. RFC 3626 and RFC 3684 describe proactive routing protocols, Optimized Link State Routing (OLSR) [14] and Topology Dissemination Based on Reverse-Path Forwarding (TBRPF) [53]. I choose to utilize AODV and OLSR as examples to demonstrate the effectiveness of the three proposed design principles. In many cases, the performance of the two channel assignment protocols, called Channel Assignment-AODV (CA-AODV) [25] and Channel Assignment-OLSR (CA-OLSR) [27], respectively, approaches that of centralized near-optimal algorithms. In addition, these two protocols exhibit much lower overhead and complexity than both centralized approaches and other existing distributed approaches.

A transmitter-based multi-channel MAC (MC-MAC) protocol is designed to work with the proposed channel assignment algorithms [26]. The combined channel assignment and MC-MAC protocol can extend the benefit of multi-channel operation to multi-hop wireless ad hoc networks. MC-MAC is essentially compatible with the single-channel IEEE 802.11 MAC due to the following reasons.

- Like the IEEE 802.11 MAC, MC-MAC is based on carrier-sense multiple access with collision-avoidance (CSMA/CA).
- MC-MAC requires only one half-duplex transceiver per host and does not require synchronization among nodes. This is exactly the same system requirement for an 802.11 node. To the best of my knowledge, the MC-MAC protocol has the minimum system requirement among all multi-channel MAC protocols proposed to date.
- Like the IEEE 802.11 MAC, MC-MAC only needs to perform medium access control, whereas most other multi-channel MAC protocols must also perform channel assignment.

Compared with the IEEE 802.11 MAC, MC-MAC has two unique design features.

1. MC-MAC does not utilize a binary exponential backoff (BEB) scheme. Instead, each node will have the same pre-determined contention window size to ensure maximum fairness in the network.
2. MC-MAC utilizes a modified virtual carrier sensing mechanism to mitigate the "exposed node" problem [8] and to improve performance.

In addition to improving performance, MC-MAC seeks to provide fair medium access to nodes in the network. As far as I know, MC-MAC is the first multi-channel MAC protocol that specifically takes fairness into consideration. Simulation results and performance analysis show that MC-MAC can improve network performance significantly. Therefore, the MC-MAC protocol and the accompanying distributed channel assignment protocol constitute an effective solution to the network capacity problem in multi-hop wireless networks.

Motivated by the effectiveness of cross-layer design for the distributed channel assignment problem, I further generalize the cross-layer design principle to other networking problems and propose a cross-layer design architecture. The architecture supports adaptation and optimization not only at a single layer but across multiple layers. Based on this architecture, prior literature is surveyed to show that the cross-layer design principle can be applied to many layers in the protocols stack to improve performance and efficiency, especially in a wireless network.

Based on the techniques that cross-layer design schemes utilize, cross-layer design can be classified into two major categories: 1) adaptive design and 2) joint layer design.

The first category of cross-layer design consists of adaptive schemes that involve one or multiple layers. Examples include adaptive rate control, power control and scheduling at the MAC layer [31, 80], interference-aware and energy-conserving routing at the network layer [34, 70], and adaptation of quality of service (QoS) requirements at the application layer.

These adaptation techniques have enhanced the ability of network protocols and applications to observe and respond to the channel variations.

However, adaptation at only one layer may adversely affect other layers and, thus, degrade overall system performance. Kawadia and Kumar give such an example in [41], where a rate adaptive MAC protocol [31] is used in conjunction with a minimum hop routing scheme called the Destination-Sequenced Distance Vector (DSDV) routing protocol [56]. Simulation results show that when combined with DSDV, the rate adaptive MAC protocol leads to performance that is worse than for the original system, i.e., a plain unmodified 802.11 MAC. This example reveals that if not designed properly, an adaptive technique could lead to performance degradation for the overall system. The proposed cross-layer design architecture offers a holistic view of the wireless system, which can help to avoid such a problem. Therefore, the proposed architecture can serve as a starting point for many adaptive designs.

The second category of cross-layer design is joint-layer designs that are based on traditional non-linear optimization techniques. Once they converge, these design techniques will reach optimal performance, which is hard to achieve using adaptive techniques. Even though non-linear optimization techniques can avoid some of the problems that adaptive techniques may encounter, joint-layer designs are often too complex to be implemented.

To summarize, this dissertation reports three major contributions. The first contribution is the proposal of three design principles for efficient distributed channel assignment and two example protocols that utilize these design principles. The second contribution is a multi-channel MAC (MC-MAC) protocol that works with the distributed channel assignment protocols. The third contribution is a cross-layer design architecture that may facilitate cross-layer designs in the future.

1.2 Organization of the Dissertation

The rest of the dissertation is organized as follows. Chapter 2 provides background and motivation for the research. Chapter 3 formulates the distributed channel assignment problem, describes design objectives and challenges for multi-channel MAC protocols, and briefly discusses research methodology. Chapter 4 discusses three design principles for distributed channel assignment and presents two example implementations. The MC-MAC protocol is described in Chapter 5. Simulation studies and capacity analysis of wireless networks are presented in Chapter 6. Chapter 7 describes a cross-layer design architecture and provides a survey of existing cross-layer design techniques. Finally, conclusions are drawn in Chapter 8.

Chapter 2

Background and Motivation

This chapter provides background and motivation for distributed channel assignment, multi-channel MAC schemes, and the cross-layer design architecture.

I begin the discussion of background by reviewing some relevant details of contention-based medium access schemes, including the IEEE 802.11 MAC Distributed Coordination Function (DCF) protocol. Since contention-based medium access schemes perform poorly in a multi-hop environment, distributed channel assignment was proposed to solve the "hidden terminal" problem. Next, the "hidden terminal" and "exposed terminal" problems are introduced. In addition, I describe a "fairness" problem that contention-based medium access protocols commonly suffer from in a wireless environment. Finally, I present the traditional TCP/IP five-layer network architecture and discuss motivation for my research on cross-layer design.

2.1 Contention-based Medium Access Schemes

A wireless ad hoc network is a purely distributed network and is often set up "on the fly". Because the coordination of mobile nodes in a highly dynamic environment is difficult, mobile ad hoc networks (MANETs) seldom use centralized medium access control (MAC) protocols to assign dedicated resources to individual nodes. Instead, contention-based protocols, such as ALOHA [3] and carrier sense multiple access (CSMA) [42], are commonly employed. These two protocols and CSMA with collision avoidance (CSMA/CA) are described below.

2.1.1 ALOHA Protocol

The basic operation of ALOHA is simple, yet elegant - stations can transmit whenever they have a packet that needs to be sent. If a collision occurs, the data packet is corrupted. The receiver can acknowledge successful receipt of the data packet. If the sender does not receive an acknowledgment within a certain timeout period, the sender assumes that there was a collision. The sender then waits a random amount of time and sends the packet again in another frame. While simple, ALOHA fails to effectively utilize channel resources. ALOHA also suffers from stability problems that can occur when a large number of stations have backlogged frames that need to be transmitted [7]. In effect, the performance of ALOHA degrades sharply when the traffic load is heavy. This performance degradation occurs mainly because all nodes transmit at will without considering transmissions at other nodes.

2.1.2 CSMA Protocols

If a node can detect whether or not other nodes are currently transmitting, it can adapt its behavior accordingly. Carrier sense multiple access is based on this idea. A family of carrier sense multiple access protocols was proposed in the 1970s by Kleinrock and Tobagi [42]. In CSMA, a node first senses the channel to make sure it is idle before starting to transmit a frame. This behavior is sometimes called “listen before talking.” If the channel is idle, the node can transmit. If the channel is busy, the node will defer transmission. The exact behavior of a node that senses a busy channel and waits leads to different versions of CSMA.

Because CSMA implements a carrier sensing scheme, it has superior performance over ALOHA, as shown in Figure 2.1.

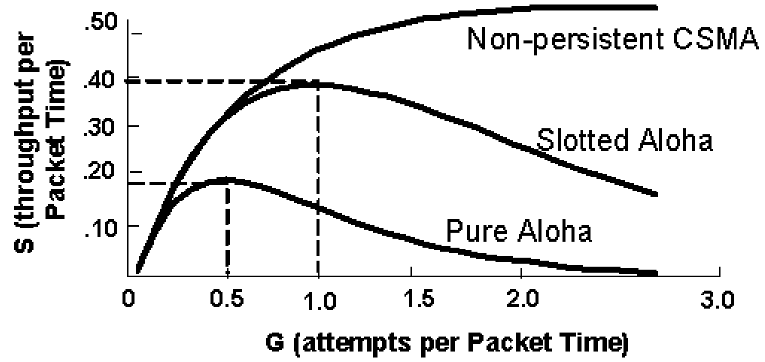


Figure 2.1: Throughput comparison of CSMA protocols and ALOHA protocols

For a one-hop wireless network where each node is within others' radio range, CSMA can achieve reasonably good performance. Variants of CSMA are widely used in networking standards. Carrier sense multiple access with collision avoidance (CSMA/CA), as used in the IEEE 802.11 MAC DCF, is such an example.

2.1.3 CSMA/CA Protocol

CSMA/CA is a commonly used protocol in wireless local area networks. CSMA/CA leverages the performance benefits of CSMA, but extends CSMA to reduce the likelihood of a collision.

The 802.11 MAC DCF protocol is a CSMA/CA based protocol that utilizes a handshake mechanism [2]. The IEEE 802.11 recommends the use of request-to-send (RTS) and clear-to-send (CTS) messages to reserve channels for transmission of most data frames, as illustrated in Figure 2.2. A node desiring to transmit first senses if the channel is free for a period of time defined as the DCF interframe space (DIFS). Then, the sender broadcasts a RTS message to reserve the medium. Upon receiving the RTS, the destination node sends back a CTS message after a Short interframe space (SIFS) period. After the channel is reversed by RTS/CTS handshake messages, data and acknowledgement (ACK) message then follow. All other nodes that overhear the RTS or CTS message must defer their transmissions for the duration of the channel reservation. This type of deferral mechanism is referred to as virtual carrier sensing or the Network Allocation Vector (NAV).

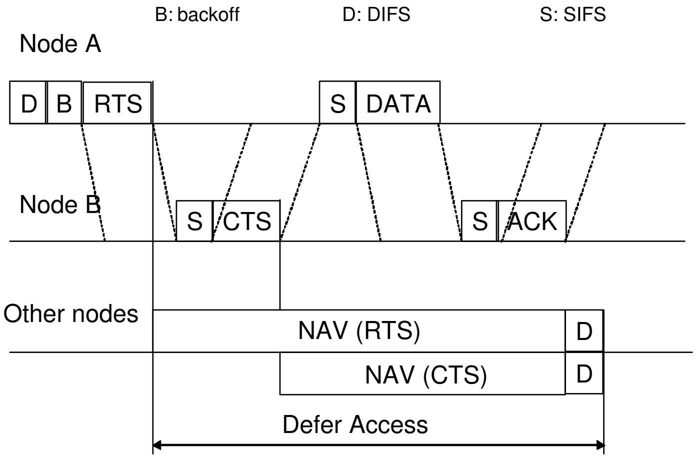


Figure 2.2: The IEEE 802.11 MAC handshake and channel reservation mechanism

The IEEE 802.11 MAC protocol can effectively mitigate the “hidden terminal” problem since it utilizes a handshake mechanism. However, because the IEEE 802.11 MAC protocol uses only a single channel, it performs poorly in a multi-hop wireless environment due to bursts of excessive utilization introduced by multiple-hop routes. The performance problem is exacerbated by the “exposed terminal” problem.

2.2 Hidden-terminal and Exposed-terminal Problems

Because radio signals attenuate over distance, simultaneous transmissions may lead to collisions at the receiver even though both senders sense the channel to be idle. A hidden terminal (node C in Figure 2.3) is a node which is out of range of a transmitter node (node A in Figure 2.3), but in the range of a receiver node (node B in Figure 2.3). Because nodes A and C are out of each other’s sensing range, they may transmit at the same time, which causes a collision at the receiver as illustrated by Figure 2.3. Such “hidden terminals” can lead to a high collision probability and cause substantial interference.

A number of MAC protocols have been proposed to mitigate the hidden terminal or hidden node problem [40, 8, 22]. The basic idea is to introduce a handshake mechanism before the actual data transmission. A sender that has a packet to transmit first sends out

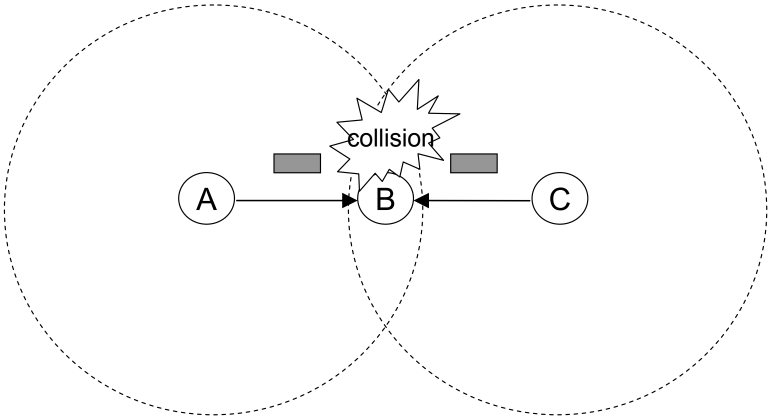


Figure 2.3: An illustration of the hidden terminal problem

a RTS to the receiver. Nodes within the sender's vicinity overhear this RTS and refrain from transmitting. Upon receiving an RTS, the receiver replies with a CTS to indicate its readiness for reception. Nodes in the vicinity of the receiver hear this CTS and also refrain from transmitting. By using the RTS/CTS handshake mechanism, collisions of data packets can be largely avoided. If a collision occurs with RTS or CTS control packets, stations involved in the collision will all backoff for some time.

Even though by utilizing a handshake mechanism, the IEEE 802.11 MAC protocol can effectively overcome the hidden terminal problem, it does not solve the "exposed node" problem. The exposed node or exposed terminal problem occurs when a node that overhears either an RTS or a CTS has to refrain from transmitting or receiving, even though its transmission may not interfere with the on-going data transmission at all. An exposed node (node C in Figure 2.4) is a node that is out of range of a receiver node (node A in Figure 2.4), but in the range of a transmitter node (node B in Figure 2.4). The dotted circles illustrate the radio range of nodes in the center of the circle. Upon detecting a transmission from node B, node C defers its transmission to node D, even though a transmission from node C does not interfere with the reception at node A. Due to the exposed node problem, the link utilization can be significantly impaired, which leads to low end-to-end throughput and high packet-delivery latency. Recent studies have shown that MAC protocols based on carrier sense multiple access (CSMA), such as IEEE 802.11's MAC protocol, are not suitable in a

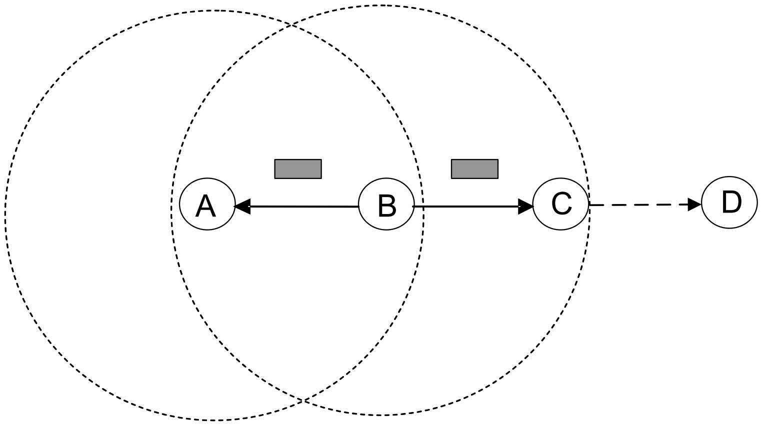


Figure 2.4: An illustration of the exposed terminal problem

multi-hop wireless environment [78].

Many schemes have been proposed to mitigate the “exposed node” problem, where the main idea is to allow the exposed node to send data packets in parallel with the sender [4, 62]. One example is the medium access via collision avoidance with enhanced parallelism (MACA-P) protocol [4]. MACA-P allows parallel data transmissions when two neighboring nodes are either both receivers or both transmitters, but a receiver and a transmitter are not neighbors. After the RTS/CTS handshake, the data transmission is delayed by a control phase interval, which allows multiple sender-receiver pairs to synchronize their data transfers, thereby avoiding collisions and improving system throughput. However, MACA-P works well only in specific types of networks. Shukla, *et al.* proposed an algorithm that is similar to MACA-P [62]. When a node identifies itself as an exposed node, it checks to see whether it has packets small enough that a new data transmissions can occur in parallel with the on-going data transmission without causing a collision with the ACK packet sent back to the sender. If the packet at the head of the queue is small enough, the node will initiate a data transmission without an RTS/CTS handshake. Note that the scheme proposed by Shukla, *et al.* works only when the data packets have different sizes.

The hidden terminal and exposed terminal problems can be largely avoided by means of distributed channel assignment schemes, where the basic idea is to assign distinct channels to any pair of nodes within radio range or two-hop range of each other [6, 23, 32]. Distributed channel assignment protocols help to achieve efficient spatial re-use of wireless channels as well as to resolve collisions and mitigate interference.

2.3 Fairness Problem

In addition to the hidden node and exposed node problems, wireless random access protocols also suffer from a fairness problem. In fact, as first investigated by Bharghavan, *et al.* [8], many variations of CSMA/CA suffer from the fairness problem.

A MAC protocol is considered fair when it provides channel access to individual nodes without giving preference to one node over the others when there is no explicit differentiation. In other words, when multiple nodes in the network are competing for channel access, the probability of each node winning the contention should be equal.

Providing fairness in a wireless ad hoc network is a challenging task for a number of reasons. First, the unfairness of wireless protocols is actually rooted in the nature of the wireless medium. In any wireless network, contention is location dependent. Nodes that are in a dense neighborhood will undoubtedly suffer more contention than nodes locate in a sparse neighborhood. Additionally, if two nodes transmit at the same effective power for the same destination, the node that is closer to the destination may succeed due to its higher received signal level. To exacerbate the fairness problem even further, the widely used binary exponential backoff (BEB) scheme is known to always favor the last node that succeeds in its transmission [8].

Secondly, there is a fundamental tradeoff between optimizing aggregate throughput and achieving fairness. For instance, allocating channel bandwidth to nodes that suffer from less contention will improve aggregate throughput, but at the expense of unfairness to nodes that suffer more contention.

Finally, because there is no central control node, it is difficult to get accurate contention or collision information. For example, in the IEEE 802.11 MAC protocol, BEB can be invoked due to the following four reasons.

1. Transmissions from two senders interfere with each other and, hence, their transmissions are not acknowledged. The absence of ACKs is treated by both senders as collisions.
2. A data packet transmitted by a sender may be corrupted due to noise and interference generated from other radio frequency sources, other networks, or from nodes that are more than two hops away from the receiver. The corrupted packet is treated as a collision by both sender and receiver.
3. In the four-way handshake model, if a sender does not receive a CTS for its RTS before a timeout, it will assume a collision has happened and start BEB. This is irrespective of the destination node. For example, the destination may defer the transmission of CTS because it detects another transmission in its vicinity.
4. Due to the *capture effect* [8] at the receiver, simultaneous transmissions may result in one success and one failure.

Out of four scenarios, only the first scenario should invoke backoff. However, BEB is invoked unnecessarily for the other three scenarios. In the fourth scenario, the node that is furthest away from the destination may suffer from starvation because, for subsequent contentions, the node that just succeeded recently will always have a higher probability of winning the contention due to its smaller contention window.

The BEB mechanism that is used in almost all wireless medium access control protocols is essentially borrowed from the wired Ethernet or IEEE 802.3 MAC protocol, where, unlike a wireless channel, is uniform for all nodes. So, the binary exponential backoff mechanism that provides reasonably fair and efficient access for a wired medium becomes the cause of unfairness in a wireless network.

Some wireless MAC protocols have been proposed to specifically address this fairness problem [8, 54, 19], where the idea is to modify the backoff counter or a “persistence probability.”

A multiplicative increase and linear decrease (MILD) scheme was proposed in the MACAW protocol [8] to address the large variation of the contention window size and the fairness problem. In MACAW, the backoff interval is increased by a multiplicative factor, specifically 1.5, upon a collision and decrease by one step upon a success, where a step is defined as the transmission time of an RTS frame. In addition, backoff timers are exchanged and copied between nodes to equalize the chance of accessing the link.

In the connection-based balanced medium access (CB-Fair) protocol [54], Ozugur, *et al.* propose a protocol that combines persistence and backoff. A p-persistent CSMA with backoff is used to differentiate flows and nodes, where the probability p is calculated by each station based on information it receives from its neighbors. The information can be empirical average contention periods. Each station then attempts to access the medium with probability p. The backoff adjustment algorithm doubles the backoff counter on each loss and halves the backoff counter on each successful transmission. CB-Fair also uses the backoff timer copying scheme similar to MACAW. However, the multiplicative nature of both the increase and decrease of backoff causes short-term unfairness similar to BEB [52]. Furthermore, the persistence algorithm tries to increase the persistence probability (p) of flows experiencing high levels of contention, which leads to artificially larger contention windows and highly inconsistent short-term behavior [52].

Fang, *et al.* proposed a measurement-based backoff algorithm to replace the BEB algorithm [19]. Each node constantly estimates its throughput and the aggregated throughput of the nodes with which it contends. Then, the station calculates a fairness index, which determines how much the node should adjust its contention window.

All three approaches assume that wireless nodes can get accurate information from their neighbors, whereas, in reality, timely and accurate estimates can be difficult to obtain [10].

If the number of available data channels is sufficiently large, the “fairness” problem can be mitigated by multi-channel MAC protocols, such as the MC-MAC protocol proposed in this dissertation. The reason is that when the number of available channels is sufficient, the channel assignment algorithms can assign channels in a way that all neighboring active nodes will be transmitting on different data channels. Therefore, data and ACK packets will not suffer any contention. In MC-MAC, only control packets such as RTS and CTS will be sent through a common control channel that is shared by all nodes in the network. Because RTS/CTS packets are small compared to typical data packets, they will suffer much less contention than the data packets.

2.4 Five-layer Network Architecture and Motivation for Cross-layer Design

In this dissertation, three design principles are proposed for efficient channel assignment. The primary principle is a “cross-layer” design principle. This cross-layer design principle may be applied to networking problems other than distributed channel assignment to improve performance and efficiency, especially in a wireless environment.

2.4.1 Five-Layer Network Architecture

Current wireless networks still adopt the traditional layered network architecture that was developed for wired networks. To simplify the construction of network protocols, the International Organization for Standardization (ISO) promulgated the Open Systems Interconnection (OSI) model [17]. The OSI model separates the functionality of the network protocols into seven layers: application, presentation, session, transport, network, data link, and physical. By hiding the operations of each layer behind well-defined interfaces, a process known as encapsulation, this layering approach facilitates the independent development of the protocols for each layer. Through the use of well-defined interfaces, the protocols in each layer of this model interact with the protocols only in layers immediately above and below its layer. In other words, a data link layer protocol may interact, perhaps by passing packets, with protocols in the network or physical layers, but not with protocols in the transport or application layers. Indeed if a network is designed properly, protocols remain blissfully ignorant of the protocols that are not immediately adjacent to them in the stack. However, in practice, the seven-layer model is commonly simplified to a five-layer TCP/IP reference model, as shown in Figure 2.5, by aggregating the presentation and session layers into the application layer.

2.4.2 Motivation for Cross-Layer Design

While this strict layered approach has greatly simplified protocol design and led to the realization of robust protocols in the Internet, it leads to performance problems in wireless ad hoc networks due to its inflexibility and static behavior. One of the great challenges in wireless networking results from the dynamic characteristic of wireless links. A wireless link is time-varying in nature and often suffers from random degradations due to noise, interference, fading and node mobility. Nevertheless, application requirements should be met despite the random variations in wireless link characteristics.

The conventional networking protocol structure, shown in Figure 2.5, is inflexible because each layer in the protocol stack is designed and operated independently and various protocol layers can only communicate in a constrained manner [1]. For instance, under the current network architecture, the application layer is not aware of the congestion level at the link layer or the signal strength at the physical layer. Thus, even when the physical layer experiences deep fading or the network is heavily congested, the application layer may keep sending data down to the lower layers, causing even more problems. Because this layered approach ignores

---

[1] It should be noted that the Transmission Control Protocol (TCP) [33] does obtain and utilize link layer information to determine the maximum segment size (MSS). So, some violations of the layered model do occur in traditional networks.

the interactions and dependencies of different network layers, it may not lead to an optimal solution.

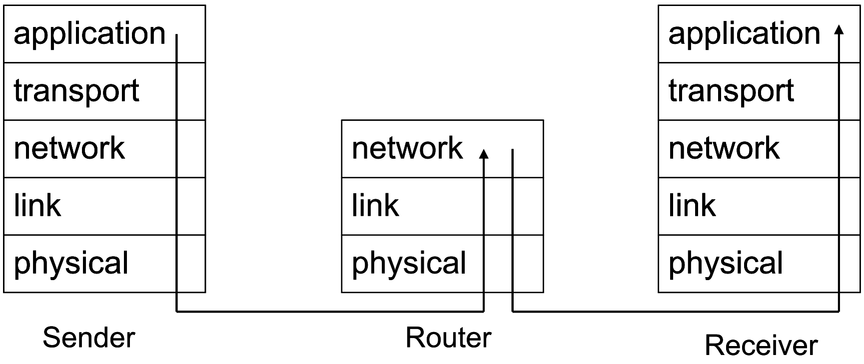


Figure 2.5: Commonly used five-layer protocol stack

In addition to being strictly layered, many networking techniques are static rather than adaptive to changing channel conditions. When no adaptation is used, optimizations can be made only at the network design phase rather than during the network's operation. Therefore, the network has to be designed assuming some pre-determined conditions, rather than adapting to actual changing conditions. This static design strategy may lead to sub-optimal solutions and inefficient use of network resources. Additionally, a layer that is optimized for a certain physical environment may not work well in a different network environment. Even though some techniques may take multiple layers into considerations, they are essentially static rather than adaptive designs. The cross-layer design techniques of interest here should optimize network performance adaptively at network run-time.

To promote and facilitate research on cross-layer design, a cross-layer design architecture is proposed in this dissertation, which provides a holistic view of the network and allows adaptation not only at a single layer but across multiple layers if necessary. This cross-layer design architecture allows information exchange among different layers and specifies the directions in which information flows from some layers to other layers. In addition, a literature review is provided to show that the proposed architecture can accommodate existing cross-layer design techniques.

2.5 Summary

In this chapter, three widely adopted contention-based MAC protocols are reviewed. They are ALOHA, CSMA, and CSMA/CA. The IEEE 802.11 MAC protocol is a CSMA/CA-based protocol that utilizes a RTS/CTS handshake mechanism. The IEEE 802.11 MAC protocol can effectively mitigate the hidden terminal problem but fails to avoid the exposed node problem. Both hidden terminal and exposed terminal problems can reduce network capacity and degrade performance. In addition, a fairness problem that is often associated with the contention-based MAC protocols can starve some unfortunate nodes in the network. The hidden terminal and exposed terminal problems provide motivation for investigating distributed channel assignment, whereas providing fairness is a second design objective of the proposed protocol.

A conventional five-layer network architecture was presented. Through the use of well-defined interfaces, the protocols in each layer of this network architecture interact with the protocols only in layers immediately above and below its layer. Even though this strictly layered approach has greatly simplified protocol design and led to the realization of robust protocols in the Internet and other data networks, it may lead to poor performance in wireless ad hoc networks due to its inflexibility and static behavior. Because of the time-varying nature of the wireless medium, cross-layer design exploring the inter-dependency among different layers becomes a promising approach.

Chapter 3

Problem Statement and Design Objectives

In this chapter, I first formulate the distributed channel assignment problem, define two performance metrics, and introduce the design objectives of distributed channel assignment. Secondly, I present design objectives and design challenges of multi-channel MAC protocols. Further, a unifying framework for cross-layer design is discussed. Finally, I briefly discuss the research methodology.

3.1 Formulation and Performance Metrics of the Distributed Channel Assignment Problem

3.1.1 Network Model and Problem Description

A wireless ad hoc network can be modelled as an undirected graph $G = \{V, E\}$, where V is the set of nodes and E is the set of edges that represent links. It is assumed that nodes use omnidirectional antennas and radio links are bidirectional. A link is assumed to exist between two nodes if and only if the two nodes are within each other's radio range.

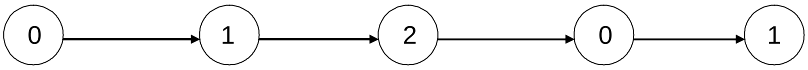


Figure 3.1: An example that illustrates the definition of a neighborhood

Two or more wireless nodes may generate *primary collisions* if they are one hop away, while *secondary collisions* can be generated by nodes that are two hops away. Such collisions can be eliminated if active nodes within a two-hop range of each other transmit on different orthogonal channels. The interference range can be defined to be the k-hop neighborhood of a node. Thus, if nodes within the k-hop neighborhood are assigned different orthogonal channels, interference can be significantly reduced. Parameter k is a user-defined neighborhood size. A larger k translates into a higher requirement of available channels, for the same signal-to-interference-noise-ratio (SINR). In general, the neighborhood size k should be chosen as large as possible, constrained by the number of available channels, to mitigate interference as well as to enable effective channel re-use in the network. Note that there is a discrepancy between "hop" distance and real physical distance. For example, a node that is k hops away may locate closer to the current node than a $k-1$ neighbor. In other words, hop distance is just an estimation of physical distance. Due to this discrepancy and the need of efficient spacial reuse, k should be chosen as large as possible.

Figure 3.1 is an example that illustrates the definition of a neighborhood. There is one active route in Figure 3.1 and the neighborhood size is $k = 2$. The arrows connecting nodes show that data packets are sent from the source node to the destination node. The values inside nodes represent channels used by the nodes. Note that channels can be spatially reused beyond the two-hop range such that no nodes within a two-hop range of each other share the same channel.

To avoid collisions as well as mitigate interference, I propose to assign distinct channels on-demand to active nodes in a k-hop neighborhood.

3.1.2 Performance Metrics

Two performance metrics, the average number of nodes in a k-hop neighborhood sharing the same channel and the average accumulated interference at active receivers, f_{CA} and R_{CA} respectively, are defined. First, the k-hop neighbors of a node v are defined as members of the set $N_k(v) = \{w \in V | l(v,w) \leq k\}$, where $l(v,w)$ is the hop distance from v to w, i.e. the minimum length of any path from v to w. Note that $N_1(v)$ is the set of directly connected neighbors of node v. $\mathcal{C}$ is defined to be a set of all available channels in the network. Given a fixed number of channels, denoted by $|\mathcal{C}|$, the first performance metric for channel assignment algorithms is the average number of nodes sharing the same channel with the designated node among this node's k-hop neighbors:

$$f_{CA} = \frac{1}{|V|} \sum_{v \in V} n_k(v). \tag{3.1}$$

Here, $n_k(v)$ is the number of nodes in $N_k(v)$ that share the same channel with node v.

If $|\mathcal{C}|$ is large enough and k is set to an appropriate value, both primary and secondary collisions can be largely avoided and harmful interference can be mitigated. Therefore, the channel assignment problem can be formulated in such a way that the goal is, given a $|\mathcal{C}|$, to minimize the average accumulated interference at each receiver. Because the transmitter-based and receiver-based channel assignments are essentially the same problem, I assume that the channel assignment scheme is transmitter based, i.e., channels are assigned to transmitters and receivers must "tune" to the channel assigned to the transmitter.

$V_t \subset V$ is defined to be the set of active transmitters and $V_r \subset V$ is defined to be the set of active receivers. Note that a node cannot be both a transmitter and a receiver at a given time, so membership in V_t and V_r is mutually exclusive, i.e. $V_t \cap V_r = \emptyset$. Let $v_{r,i} \in V_r$ be a particular receiver and $v_{t,j} \in V_t$ be a particular transmitter. If $v_{t,j}$ is transmitting on the same channel on which $v_{r,i}$ is receiving, but $v_{t,j}$ is not the intended transmitter to $v_{r,i}$, then transmitter $v_{t,j}$ will interfere with receiver $v_{r,i}$. The power associated with the interference, $P(v_{t,j}, v_{r,i})$, is the received power at node $v_{r,i}$, which is a function of the effective transmitter power of node $v_{t,j}$ and the distance between nodes $v_{t,j}$ and $v_{r,i}$, and is determined by the path loss model. The average interference at active receivers is determined as follows.

$$R_{CA} = \frac{1}{|V_r|} \sum_{v_{r,i} \in V_r} \sum_{v_{t,j} \in V_t \setminus v_{t,T(i)}} \left[P(v_{t,j}, v_{r,i}) \cdot S(v_{t,j}, v_{r,i}) \right]. \tag{3.2}$$

Node $v_{t,T(i)}$ is the intended transmitter for receiver $v_{r,i}$ and is excluded from the inner summation of interference sources. The term $S(v_{t,j}, v_{r,i})$ indicates the relation between the channels used by transmitter $v_{t,j}$ and receiver $v_{r,i}$. Function $S(v_{t,j}, v_{r,i}) = 0$ if $v_{t,j}$ and $v_{r,i}$ are using different (strictly orthogonal) channels and $S(v_{t,j}, v_{r,i}) = 1$ if $v_{t,j}$ and $v_{r,i}$ are using the same channel. In code-division multiple access (CDMA) systems, different channels with non-orthogonal codes may be used, in which case $0 < S(v_{t,j}, v_{r,i}) < 1$. Table 3.1 summarizes the notation used.

3.1.3 Design Objective of Distributed Channel Assignment

Based on the performance metrics introduced above, the design objective is presented. The primary design goal of distributed channel assignment is to minimize the maximum number of nodes sharing the same channel with any designated node $v_{t,j} \in V_t$ among this node's k-hop neighbors.

Minimize:

$$\max n_k(v_{t,j}), \qquad \forall\ v_{t,j} \in V_t \tag{3.3}$$

subject to:

$$\frac{P(v_{t,T(i)}, v_{r,i})}{\sum_{v_{t,j} \in V_t \setminus v_{t,T(i)}} \left[P(v_{t,j}, v_{r,i}) \cdot S(v_{t,j}, v_{r,i}) \right] + N_0} \geq \beta, \qquad \forall\ v_{r,i} \in V_r. \tag{3.4}$$

Here, N_0 is the assumed additive white Gaussian noise (AWGN) noise and β is the minimum SINR required for a successful packet reception. In Equation (3.4), note that node $v_{t,T(i)}$ is the intended transmitter for receiver $v_{r,i}$ and is excluded from the summation of interference sources.

Table 3.1: Notation Used in Section 3.1

Notation	*Comments*
$\mathcal{C}$	A set of available channels
V	A set of nodes in the network
E	A set of edges that represent radio links
$v \in V$	A node in the network
$n_k(v)$	Number of k-hop neighbors sharing the same channel with node v
$V_t \subset V$	A set of active transmitters
$V_r \subset V$	A set of receivers
$v_{t,j} \in V_t$	A particular transmitter
$v_{r,j} \in V_r$	A particular receiver
$v_{t,T(i)}$	The desired transmitter
$P(v_{t,j}, v_{r,i})$	The power level of the received signal from transmitter $v_{t,j}$ to receiver $v_{r,i}$
$S(v_{t,j}, v_{r,i})$	The relation between the channels used by transmitter $v_{t,j}$ and receiver $v_{r,i}$
β	The desired SINR threshold for successful packet reception
N_0	Additive white Gaussian noise (AWGN)

The constraint implies that the accumulated interference generated by active transmitters sharing the same or interfering data channels as the designated transmitter plus noise N_0 should be less than a certain threshold β to ensure that the receiver can decode the data packet successfully.

To meet the primary design goal and to obey the design constraint, a channel assignment protocol should distribute available channels with any pre-defined value of k in a way that the maximum number of transmitters that share the same data channel is minimized. Meanwhile, the same set of channels should be re-used in a way that the accumulated interference generated on any particular data channel is below a certain threshold.

3.2 Design Objectives and Challenges of Multi-Channel MAC Protocols

To improve the capacity of wireless ad hoc networks, a practical distributed channel assignment protocol is closely coupled with a multi-channel MAC protocol. In this section, I will discuss design objectives and design challenges of multi-channel MAC protocols.

3.2.1 Design Objectives and Performance Metrics

A multi-channel MAC usually consists of two major management components: channel assignment and medium access control. Channel assignment determines which channel is to be used by which node, while medium access control is responsible for coordinating the access of a particular channel, resolving contention and collisions on that channel.

The most important goals of multi-channel MAC design are twofold: to achieve a high channel throughput and to reduce delay. These goals are achieved by maximizing spatial re-use of multiple channels and minimizing contention and collisions on each individual channel. The channel assignment component is designed to utilize as many channels as possible within a neighborhood and to keep channel conflicts to a minimum. Meanwhile, the medium access control component guarantees that even when two neighboring nodes select the same channel, contention and collisions on the channel can be effectively controlled and resolved.

In addition to the most important design goals, there are secondary design objectives such as fairness, energy efficiency, scalability, and quality of service (QoS) support. To date, most protocols only consider the two most important design goals. However, with increasing interest in this field, secondary design objectives are expected to be considered in multi-channel MAC design in the near future. In my proposed MC-MAC protocol, a secondary design goal, i.e. fairness, is also considered. To provide fairness, a MAC protocol needs to provide channel access to individual nodes without giving preference to one node over others when there is no explicit differentiation.

To evaluate the performance of multi-channel MAC protocols, various performance metrics can be used. The following two performance metrics are used to evaluate the performance with respect to the two most important design objectives.

- *Aggregate throughput.* Aggregate throughput is the throughput measured over all available channels, where throughput is defined as the fraction of the channel capacity used for data transmission. Therefore, to achieve a high aggregate throughput, a multi-channel MAC protocol should achieve high channel utilization on each single channel as well as effectively utilize all available channels.

- *Delay.* Traditionally, delay is defined as the average time spent by a packet in the transmit queue, including queuing delay and transmission delay. Some multi-channel MAC researchers also use end-to-end delay as a performance measure [63], which consists of delays accumulated over multiple hops from a source node to a destination node.

3.2.2 Design Challenges

While wireless ad hoc networks can extend network coverage and potentially increase network capacity, they also impose unique challenges on MAC protocol design. The first and foremost challenge stems from the ad hoc nature of wireless ad hoc networks. In the absence of fixed infrastructure that characterizes traditional wireless networks, control and management of wireless ad hoc networks have to be *distributed* across all nodes. Distributed medium access control is a much more challenging problem than centralized medium access control. For multi-channel MAC protocols, distributed channel selection or channel assignment adds another level of difficulty.

The second challenge is due to the multi-hop transmissions in wireless ad hoc networks where nodes are not all directly connected. Because nodes may not necessarily be within each other's radio range, packets have to be relayed from one node to another before they can reach the destination. In wireless networks, radio signals attenuate over distance. Therefore, simultaneous transmissions may lead to collisions at the receiver even though both senders sense the channel to be idle. Due to the inefficiency of carrier sensing in multi-hop wireless networks, the hidden node problem and exposed node problem may occur [8]. Hidden nodes can lead to a high collision probability and cause substantial interference, while exposed nodes can significantly reduce the link utilization. Both problems lead to low end-to-end throughput and high packet-delivery latency.

For multi-channel MAC protocols, both the hidden node problem and the exposed node problem may still exist, for instance, when neighboring nodes pick the same data channel for transmitting or receiving. The hidden node problem occurs in a multi-channel system when a node picks a data channel to transmit without knowing that another node is transmitting on the same channel [63]. To solve the multi-channel hidden node problem, nodes need to be either synchronized [63] or equipped with multiple interfaces, where one interface always listens on a common control channel [76]. In my proposed protocols, the hidden node problem is mitigated by exchanging routing control messages embedded with channel assignment information. In addition to the multi-channel hidden node problem and the exposed node problem, nodes that utilize a multi-channel MAC protocol may suffer from a "deafness" problem. The deafness problem occurs when two neighboring nodes choose different channels for transmitting and receiving. Therefore, even though the two nodes are within each other's radio range, they cannot detect each other. The deafness problem can be mitigated by allowing neighboring nodes share a common channel.

The third challenge is introduced by node mobility, when present, in wireless ad hoc networks. When the network topology changes, nodes assigned to use identical channels (i.e., spacial reuse in the old topology) may become neighbors and interfere with each other in the new topology. Accordingly, a multi-channel MAC may need to reassign channels to mobile nodes and adjust the medium access control scheme accordingly.

In addition to the above challenges that are unique to wireless ad hoc networks, multi-channel MAC design also faces challenges due to the error-prone nature of wireless channels. Link-layer retransmission schemes are often used in wireless MAC protocols to make the wireless link more reliable.

3.3 Research Methodology

Three design principles for the distributed channel assignment problem, two example implementations, and a multi-channel MAC protocol are presented in this dissertation. Through analytical study, I show the properties of the proposed protocols. The performance of proposed protocols is evaluated through ns-2 simulations.

The cross-layer design principle can be generalized to other network design problems to achieve better performance and higher efficiency. Even though many cross-layer design techniques have been proposed over the years, little effort has been directed towards providing a unifying framework for cross-layer design in general. I present a cross-layer design architecture that not only is helpful for classifying the current state of the art but also can guide future cross-layer design. A comprehensive review of current cross-layer design techniques further validates the proposed architecture.

3.4 Summary

In this chapter, the distributed channel assignment problem is formulated and two performance metrics are introduced. In addition, I discuss the design goals and challenges of distributed channel assignment and multi-channel MAC protocols. The design principles and protocols of distributed channel assignment are presented in Chapter 4, while MC-MAC is discussed in Chapter 5.

In addition, I briefly discuss the research methodology, which is combining analytical study with simulation evaluation. Simulation results are presented in Chapter 6.

Chapter 4

Design Principles of Distributed Channel Assignment

In this chapter, three design principles for distributed channel assignment are presented. Additionally, two example uses of these design principles are introduced.

4.1 Design Principles

The first and primary principle is a "cross-layer" design principle, where channel assignment is combined with ad hoc routing. This cross-layer design principle is motivated by the fact that both the channel assignment algorithm and the ad hoc routing algorithm must be invoked when there is a change in the network topology. In addition, transmitting channel information through routing control messages can greatly reduce the communication complexity of channel assignment protocols. For instance, a recently proposed channel assignment algorithm has a communication complexity of $O(d^2 \cdot |V|)$, where d denotes the maximum number of one-hop neighbors that a node can have [23]. This complexity implies that whenever there is a topology change, up to $O(d^2 \cdot |V|)$ messages will be exchanged in the network. Such high complexity prevents the protocol from being implemented in a realistic scenario. However, by utilizing the cross-layer design principle that I propose, two proposed distributed channel assignment algorithms have a communication complexity of only $O(1)$, since channel information is carried exclusively through routing control messages.

The second design principle states that in reactive routing protocols, channels should be assigned only to active nodes, while in proactive routing protocols, active nodes have a higher priority of picking distinct channels over inactive nodes. In a network that utilizes reactive routing protocols, before a node has a valid route, it cannot transmit or receive data packets and, thus, does not need a channel. This type of node is an “inactive” node. A node with a valid route is called an “active” node. Most existing channel assignment schemes assign channels to *all* nodes in a wireless network, regardless of whether they are active or inactive [6, 23, 32]. If not all nodes in a network are active at the same time, the existing schemes will require more wireless channels than necessary. By assigning channels on-demand only to active nodes or giving higher priority to active nodes, the resulting protocols can potentially reduce the number of channels required in a wireless network, since the number of required wireless channels is generally proportional to the number of active nodes rather than the total number of nodes.

Finally, distinct channels should be assigned in a way that collisions and interference can be avoided as much as possible. Many existing channel assignment schemes try to resolve only secondary collisions, whereas primary collisions can also reduce channel utilization. Moreover, interference can corrupt data packets and reduce throughput. Instead of assigning distinct channels to a node and its two-hop neighbors, which is what existing algorithms do, I propose to assign distinct channels to any nodes within a k-hop neighborhood. Neighborhood size k is a user-defined constant that can be set to a value that is appropriate for specific types of networks, modulation, error control coding, and applications. By maintaining distinct channels within a k-hop neighborhood, network capacity might be greatly improved.

4.2 Example Protocols

The proposed design principles can be applied to both reactive routing algorithms, such as the Dynamic Source Routing (DSR) [38] and Ad-hoc On Demand Distance Vector (AODV) [57] protocols, and proactive routing protocols, such as the Optimal Link State Routing (OLSR) protocol [14]. Because proactive routing protocols do not set up routes on-demand, channel information is maintained for a node's k-hop neighbors at all times, where k is defined as the size of the neighborhood. However, the active nodes in the neighborhood have priority when picking distinct channels over nodes that are not active at the current time. Additionally, the channel assignment protocol is invoked on-demand whenever a node detects a channel conflict. Because the channel assignment algorithm is combined with a routing protocol, the algorithm is not constrained to only one-hop neighbors of a node. Instead, a node can obtain channel information for any node k hops away and then use this information to select a channel for itself. Moreover, by assigning distinct channels to neighbors within a node's interference range, the proposed protocols not only solve the hidden terminal problem, but also significantly reduce packet errors caused by interference from other transmitters.

The details of different routing protocols may differ, but most of the essential design principles remain the same. I use AODV, a reactive routing protocol, and OLSR, a proactive routing protocol, as examples to illustrate the three design principles.

AODV is a reactive routing protocol, which means nodes do not maintain up-to-date routes to all destinations at all times. Instead, a node initiates a route discovery procedure by broadcasting a Route Request (`RREQ`) message only when it has packets for the destination node and it has no valid route to the destination. OLSR is a proactive routing protocol. As the name suggests, OLSR is essentially a link state routing protocol with an optimized flooding method that can reduce the communication overhead [14]. OLSR minimizes the overhead of control packet flooding by using only some selected nodes, called multipoint relays (MPRs), to retransmit control messages. Because OLSR allows each node to have complete up-to-date topology information (possibly with some delay due to delays in message exchange) for the network based on periodically exchanging control messages, OLSR can be closely coupled with a channel assignment protocol, without causing significant modifications to the protocol itself.

First, I will present a combined channel assignment and AODV scheme, namely, the Channel Assignment AODV (CA-AODV) protocol, and two of its extensions: the Enhanced k-hop CA-AODV (Ek-CA-AODV) protocol and the Enhanced 2-hop CA-AODV (E2-CA-AODV) protocol.

CA-AODV aims to assign distinct channels to any node that is on the same active route and is within k hops of the current node. CA-AODV does not introduce any extra control messages. Instead, it utilizes AODV's own routing control messages, such as `Route Request` and `Route Reply`, exclusively. Ek-CA-AODV introduces an extra control message for channel assignment in addition to utilizing routing control messages. Ek-CA-AODV seeks to assign distinct channels to any nodes within k hops of the current node, regardless of whether they are on the same active route or not. Thus, with a little extra complexity, Ek-CA-AODV can achieve better performance than plain CA-AODV. E2-CA-AODV is a variant of Ek-CA-AODV. Like CA-AODV, E2-CA-AODV does not introduce any extra control messages. E2-CA-AODV utilizes AODV `Route Request` and `Route Reply` routing control messages. Additionally, E2-CA-AODV utilizes AODV `HELLO` messages to convey channel information to a node's neighbors in a two-hop range. Therefore, E2-CA-AODV can help both resolve collisions and mitigate intra-flow interference.

Secondly, a combined channel assignment and OLSR (CA-OLSR) scheme is presented. CA-OLSR exclusively utilizes OLSR's own routing control messages, such as `HELLO` message and topology control (`TC`). Because, in a proactive routing protocol, each node has complete and up-to-date knowledge of the network topology, CA-OLSR can assign distinct channels to any active nodes within k hops of the current node.

The proposed algorithms work with a multichannel carrier sense multiple access (CSMA) MAC protocol that will be introduced in the next chapter. All nodes use the same common channel for control messages, but may use different data channels. The algorithms presented here assign channels based on transmitters, but they can be modified slightly to assign channels based on receivers as well, since transmitter-based channel assignment and receiver-based channel assignment are essentially the same problem.

4.3 Combined Channel Assignment and AODV Protocol (CA-AODV)

Like the original AODV routing protocol and many other reactive protocols, CA-AODV has two phases: route discovery and route reply. The route discovery phase is initiated only when a source node needs to send one or more packets to a destination to which the source does not have an up-to-date route. The route reply phase is initiated when the destination node or an intermediate node that has a valid route to the destination receives the Route Request (`RREQ`) message and generates a Route Reply (`RREP`) message.

4.3.1 Route Discovery and Route Reply

Route Discovery

At the beginning of the route discovery phase, the source node broadcasts a route request that carries the index of its channel in addition to fields that are necessary for routing. If the source node does not yet have a channel, it randomly picks a channel from $\mathcal{C}$, i.e. the set of all channels that can be used in the network. Any node that receives the `RREQ` message updates its next-hop table entries with respect to preceding nodes in the path back to the source. Each table entry consists of both the route and the indices of channels that have been taken, so far, by the node's k-hop neighbors on the same route. If the node has no channel assigned to it, it updates its available channel set, denoted by $\mathcal{A} \subset \mathcal{C}$, by marking the channel taken by the preceding k (or fewer if the route is not k hops long) nodes on the path as unavailable. Then, it randomly picks a channel from the set of available channels. If a node does not have a valid route to the destination, it adds both its own channel index and the channel indices of its previous $k-1$ upstream neighbors to the `RREQ` message and rebroadcasts the `RREQ` message to its neighbors.

A combined route request and code assignment procedure is illustrated in Figure 4.1. Here, neighborhood size k is assumed to be 2 and the total number of available channels is six. At the beginning of the route discovery procedure, the source node (node A in Figure 4.1) broadcasts a route request (`RREQ`) packet that contains source id, destination id, time-to-live (TTL), and its own channel index. Since the source node A does not have a channel yet, it randomly picks a channel (channel 2) from a complete set of channels, denoted by $\mathcal{C}$.

Any node that receives the `RREQ` from the source node updates its next-hop table entries with respect to the source. Meanwhile, it should update its available channel set, denoted by $\mathcal{A}$, by marking the channel taken by the source as unavailable. If the node has no channel assigned to it, it randomly picks one from the set of available channels. For instance, upon receiving `RREQ` from node A, node E marks channel 2 as unavailable: $\mathcal{A} = \mathcal{C} \backslash \{2\}$. Then, node E randomly picks a channel (channel 5) from $\mathcal{A}$. If a node does not have a valid route to the destination, it will add its own channel index and the channel indices of its $(k-1)$-hop neighbors to the `RREQ` and rebroadcast the `RREQ` to its downstream neighbors. Note that the `RREQ` broadcast by node E now contains two channel indices: its own channel index 5 and node A's channel index 2. Each of the following intermediate nodes updates its next-hop table entry, updates the available channel set $\mathcal{A}$, updates the `RREQ` with its own channel index and the channel index of its one-hop upstream neighbor, and re-broadcasts the `RREQ`. If a node receives multiple `RREQ` messages with the same sequence number and the same source node, it only accepts the `RREQ` message that arrives first and discards the rest. For instance, even if node D receives two `RREQ` messages, one from node E and the other one from node B, node D only accepts the `RREQ` message from node E because it arrives first.

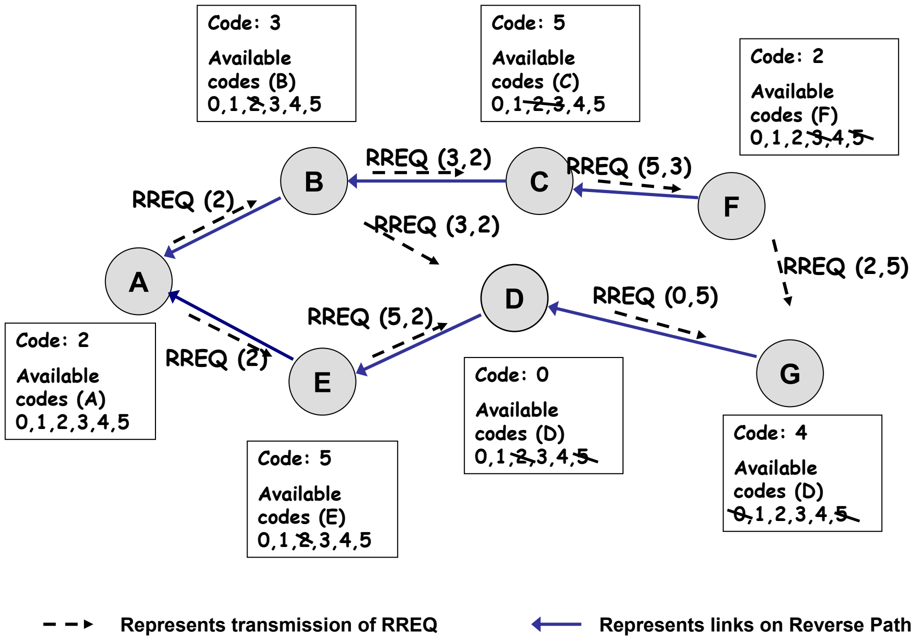


Figure 4.1: A combined route request and channel assignment procedure

Route Reply

Upon receiving a `RREQ` message, the destination node or a node that has a valid route to the destination will unicast a route reply (`RREP`) message back to the source. Upon receiving the `RREP` packet, each node along the route back to the source updates its next-hop table entries with respect to the nodes along the path to the destination node.

The combined route reply and code assignment procedure is illustrated in Figure 4.2. The unicasted `RREP` should contain the initiating node's own channel index. For instance, in Figure 4.2, the destination node G unicasts a `RREP` that contains its own channel index, i.e. 4. Upon receiving the `RREP` packet, each node along the `RREP` route updates its next-hop table entries with respect to the destination node and update the `RREP` packet with its own channel index and the channel index of its one-hop upstream neighbor.

After the route has been established, each node along the route should have a channel that is different from any of its k-hop neighbors on the same route, assuming there are sufficient channel. A route expires if it is not used or reactivated for a certain period of time.

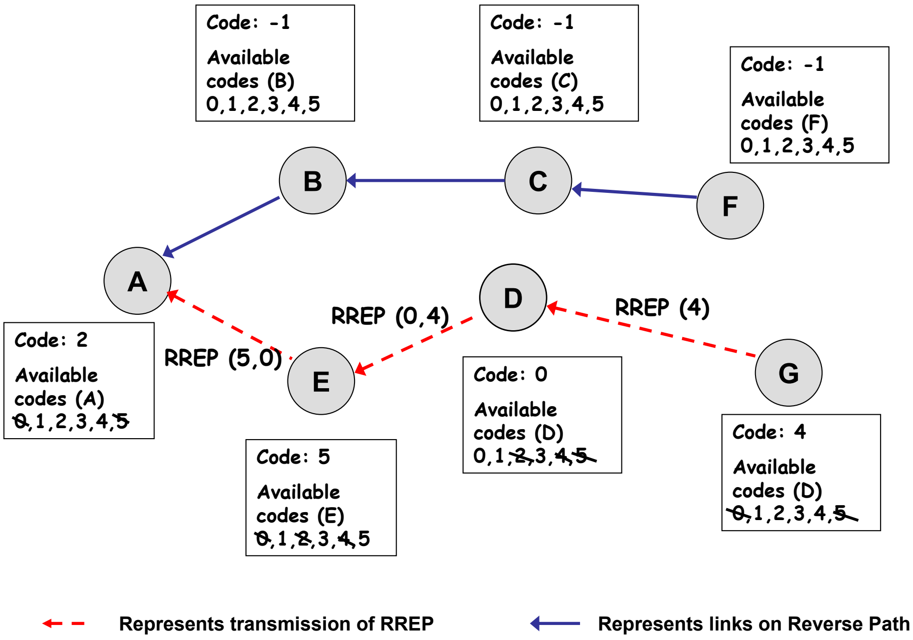


Figure 4.2: A combined route reply and code assignment procedure

The entry that corresponds to this route will then be deleted from the routing table and the channels taken by k-hop neighbors on the expired route will be marked as available again. When a node becomes an inactive node due to route expiration, it changes its own channel index to INVALID, i.e. -1 in Figure 4.2.

Ideally, when there are several active routes that traverse the same node, this node should be able to convey all of its known one-hop neighbor channel information to other nodes on the same routes. However, if a node informs others about current active routes, it must broadcast an expiration message to its neighbors whenever a route expires (or periodically broadcast renewal messages to indicate that the route is still active). Such broadcasts require extra control messages in addition to AODV's routing messages and, thus, introduce overhead. Therefore, in CA-AODV, the combined routing and channel assignment procedure is performed independently for each route that is discovered.

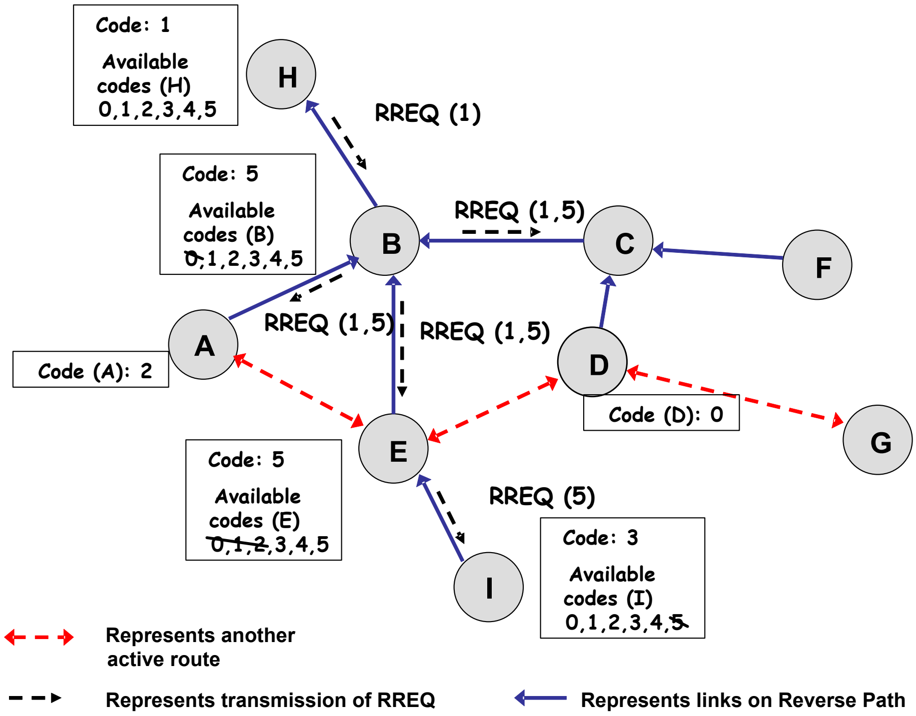


Figure 4.3: An illustration of two active routes that cross one node

4.3.2 Resolution of Channel Conflicts

Because routes are independently established in CA-AODV, channel conflicts may occur if the path of an active route crosses other routes. Figure 4.3 illustrates a case where two active routes cross at the same node and there are six available wireless channels.

In Figure 4.3, node E lies on two active routes. One active route has been established and the other one is being set up. On the route being set up, node H is the originator or the source and node I is the destination. Because node B does not know that node E is using channel 5, node B picks the same channel as node E. However, because node E is already on an active route, it cannot change its own channel. A mechanism is needed to inform node B to change its channel. This type of conflict can be resolved by the combined route reply and channel assignment procedure shown in Figure 4.4. When node E sends a `RREP` to node B, it indicates that channel 5 has already been taken. Upon receiving this `RREP`, node B marks channel 5 as unavailable and randomly picks another channel from the available channel set.

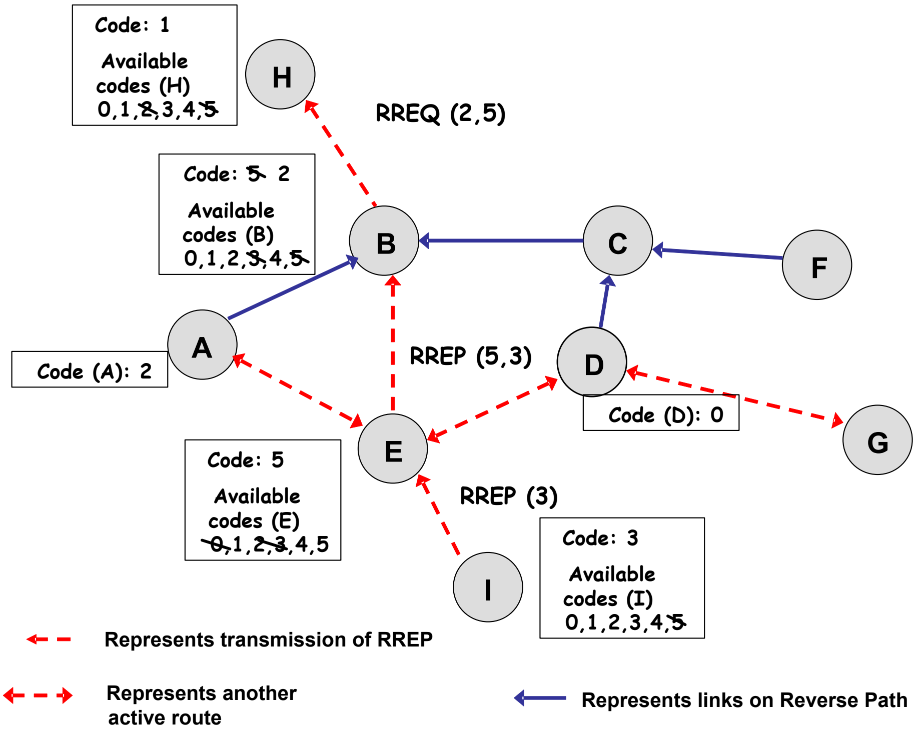


Figure 4.4: When two active routes cross at one node

In the example of Figure 4.4, node B picks channel 2.

As suggested by this example, the CA-AODV protocol can still work well when there are multiple active routes traversing a node. Even though CA-AODV introduces very low control overhead, the performance of CA-AODV can be improved, especially when multiple active routes co-exist in a k-hop neighborhood.

If performance is desired over efficiency and sufficient channels are available, the protocol can be modified in such a way that distinct channel assignment for all nodes within the mutual interference range can be achieved, regardless of how many active routes are in the network, assuming a sufficiently large number of available channels. However, extra control messages have to be sent in addition to `RREQ` and `RREP` messages, thus adding to communication overhead and the complexity of the protocol.

4.4 Enhanced k-hop Channel Assignment and AODV Protocol (Ek-CA-AODV)

CA-AODV avoids channel collisions on a per route basis. Thus, it works well only when there are few active routes co-existing in a k-hop neighborhood. To improve the performance of CA-AODV, an extension of CA-AODV, called Ek-CA-AODV, is proposed.

Ek-CA-AODV introduces an extra control message called `ChannelTaken`. Most of the operations of Ek-CA-AODV are similar to those of CA-AODV except for the operation involving the extra control message `ChannelTaken`. If a node on an established route detects that a new route in the neighborhood is being set up, it broadcasts a `ChannelTaken` message that carries its own channel index. The TTL of the `ChannelTaken` message is set to k to ensure that the `ChannelTaken` message is broadcast only to the current node's k-hop neighbors. Upon receiving a `ChannelTaken` message, any node will update its next-hop neighbor table and the available channel set $\mathcal{A}$. If a channel conflict is detected by a node that is not yet on an established route, this node sets a `channelConflict` flag. When receiving a `RREP` message, a node checks to see whether the `channelConflict` flag is set. If so, the node will randomly pick another channel from the channel set $\mathcal{A}$. Note that nodes on active routes just ignore channel conflicts because the channel conflicts are caused by inactive nodes that are not on active routes but are trying to discover a new route. Therefore, an inactive node updates its channel whenever a channel conflict occurs.

Through `ChannelTaken` messages, channels taken by nodes on established routes can be conveyed to other nodes in the network. Therefore, conflicting channels within the k-hop neighborhood can be largely avoided, provided that the number of available channels is sufficient.

To allow sufficient time for `ChannelTaken` messages to propagate to all nodes within the k-hop range, the destination node or a node that has a valid route to the destination should wait for a period of time, denoted by W_t, before sending back the `RREP` message. W_t

is related to both k and t_p, where t_p is the per hop propagation time. If W_t is set to a large value, it will increase routing delay. If W_t is chosen to be too small, `ChannelTaken` messages may not propagate to nodes on the route being established and thus adversely affect the performance of channel assignment. In the simulations that I have conducted, W_t is set to be $2k \times t_p$.

4.5 Enhanced 2-hop Channel Assignment and AODV Protocol (E2-CA-AODV)

E2-CA-AODV is another extension of CA-AODV. Because CA-AODV only considers collisions and interference on the same route, it does not perform well when multiple active nodes on different routes are co-located in the same k-hop neighborhood. Ek-CA-AODV tries to take both intra-flow interference and inter-flow interference into consideration, at the cost of higher communication overhead and additional delay W_t in route setup. E2-CA-AODV has the same communication overhead as CA-AODV, but it seeks to assign distinct channels to active nodes on the same route and active nodes within the two-hop neighborhood. Therefore, E2-CA-AODV can both avoid collisions and mitigate intra-flow interference.

In addition to `RREQ` and `RREP` messages, E2-CA-AODV utilizes another AODV routing control message, the `HELLO` message. `HELLO` messages are exchanged among one-hop neighbors. If a node is an active node, it will indicate its assigned channel and a *NodeNumber* in the `HELLO` messages that it broadcasts to its one-hop neighbors. Each time a node chooses a data channel, it updates its *NodeNumber* by randomly pick a very large number. This is to minimize the likelihood that two or more neighboring nodes choose the same random number to be their *NodeNumbers*. `HELLO` messages also carry the channel and *NodeNumber* information of a node's active one-hop neighbors. Upon receiving a `HELLO` message from a one-hop neighbor, a node will update its available channel set $\mathcal{A}$ by removing the channels taken by active neighbors from the set. If there is a channel conflict involving two active nodes, the node with the smaller node number will retain its channel while the other node shall randomly pick another data channel from the set $\mathcal{A}$. The node that updates its own data channel will inform its neighbors in the next `HELLO` message that it sends.

The other operations of E2-CA-AODV, such as Route Request and Route Reply, are similar to those of CA-AODV. E2-CA-AODV utilizes three types of routing control messages, namely `RREQ`, `RREP`, and `HELLO` message. Because E2-CA-AODV allows channel information to be exchanged among two-hop neighbors, it can successfully avoid primary and secondary collisions that involve active nodes on different routes.

4.6 Combined Channel Assignment and OLSR (CA-OLSR) Protocol

OLSR is a proactive routing protocol. As the name suggests, OLSR is essentially a link state routing protocol with an optimized flooding method that can reduce communication overhead [14]. OLSR minimizes the overhead of control packet flooding by using only selected nodes, called multipoint relays (MPRs), to retransmit control messages. Because not all nodes are involved in the flooding of control traffic, OLSR can reduce the number of retransmissions required to diffuse a control message through the network. Secondly, OLSR requires only partial link state to be flooded to provide shortest path routes. Therefore, the size of control packets can be reduced. Each node selects its MPR set among its one hop neighbors in such a way that the set covers all the two-hop neighbors. For instance, the black nodes in Figure 4.5 are MPRs selected by the node in the middle. Once a node is selected as a multipoint relay, it not only retransmits routing control messages, but also serves as an intermediate node on routing paths. A route is essentially a sequence of hops through the multipoint relays from source to destination.

Two types of control messages are defined in OLSR: `HELLO` messages and topology control (`TC`) messages. To select MPRs and calculate its routes to all known destinations through

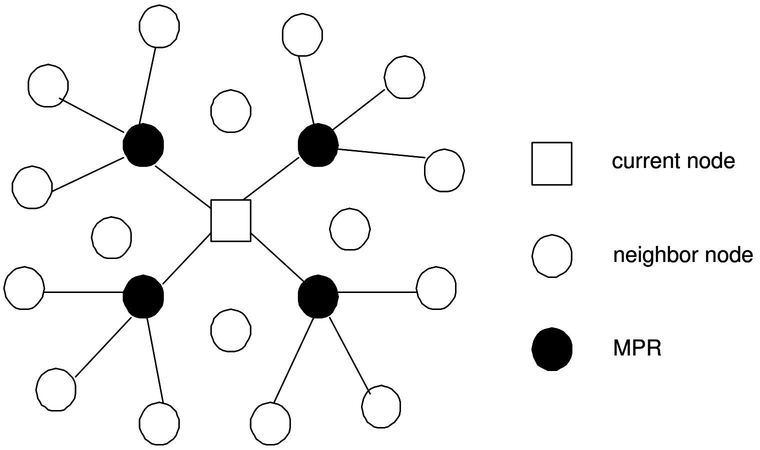


Figure 4.5: Illustration of multi-point relays

the MPRs, a node must first detect its one-hop and two-hop neighbor nodes by periodically broadcasting `HELLO` messages, which have a TTL value of 1. `HELLO` messages permit each node to learn the topology of its neighbors up to two hops away. Based on this information, each node in the network independently selects its own set of MPRs that covers all the two-hop neighbors. MPRs of a given node are then declared in the subsequent `HELLO` messages so that the information can reach the MPRs themselves. A neighbor table and an MPR selector table are constructed by each node using information obtained from the `HELLO` message.

A `TC` message is sent periodically by each node to declare its MPR selector set that consists of the list of neighbors who have selected the sender node as an MPR. `TC` messages are forwarded by nodes in the MPR set to the entire network. Each node maintains a topology table to record the topology information obtained from the `TC` messages. A `TC` message is larger than a `HELLO` message and is sent out less frequently than the `HELLO` message. A routing table is constructed at each node based on the information contained in the neighbor table and the topology table. The calculation of the routing table can follow any standard link state algorithm. An example procedure is given in RFC 3626 to explain how to calculate the routing table [14].

The basic ideas of CA-OLSR are to utilize routing control messages to exchange channel information and to assign distinct channels to active transmitters within a k-hop neighborhood. Because MPRs are the nodes that act as intermediate routes for other nodes, it is intuitive to give MPRs higher priority when the number of channels is insufficient to ensure no interference by all nodes. However, at any time, not all MPRs are active. Moreover, nodes that do not act as MPRs may become active source nodes. To consider these situations, I propose to identify active nodes within a certain time period and then assign a higher priority to active nodes when there is a channel conflict. Because `HELLO` messages are sent more frequently and can carry channel information up to two hops away, we choose `HELLO` messages to carry channel information. The neighborhood size in CA-OLSR is defined to be $k = 2$, based on the assumption that interference range is twice the radio range.

At initialization, each node in the network randomly chooses a channel from a set of all available channels, denoted by $\mathcal{C}$, and chooses a large random number, i.e. *NodeNumber*. Then, each node sends out a `HELLO` message that carries its own channel index and its *NodeNumber*. After a node v has sensed its one-hop neighbors, the `HELLO` message will contain a list of its one-hop neighbors $N_1(v)$. A node should then include the channel and *NodeNumber* information of its active neighbors in future `HELLO` messages. Nodes detect active neighbors by listening on the control channel. Neighbors that exchange RTS and CTS control messages in one hello interval are considered active neighbors, which are then indicated in the `HELLO` messages sent in the next hello interval. If a node has selected its MPR set $M \in N_1(v)$, it should also indicate its MPRs in the `HELLO` message. Upon receiving `HELLO` messages from neighboring nodes, a node first checks to see whether there is a neighbor set contained in the message. If so, the node builds an available channel list, $\mathcal{A}$, by marking channels that are taken by active neighbors as unavailable.

If there is a channel conflict between the current node and an active neighbor, the current node should choose another channel from the available channel set $\mathcal{A}$. This is to ensure that active nodes have higher priority to obtain distinct channels than other nodes when the number of available channels is fewer than the number of nodes in the two-hop neighborhood. If there is a channel conflict between two active nodes, the node with the smaller *NodeNumber* retains its channel while the other node should mark the channel-in-conflict as unavailable and randomly pick a new channel from its updated available channel set $\mathcal{A}$. The same procedure applies when two inactive nodes have a channel conflict. In general, active nodes have higher priority over in-active nodes. Within the set of active or inactive nodes, the nodes with smaller *NodeNumbers* have higher priority. Some existing channel assignment algorithms resolve channel collisions based on node IDs. Because node IDs do not change during the network lifetime, the nodes with lower node IDs may be favored over the nodes with higher IDs, which may cause unfairness. Therefore, we proposed to resolve collisions based on *NodeNumbers*, which are updated each time when a node updates its data channel. Because *NodeNumber* is chosen from a large set of numbers, the likelihood that two nodes choose the same *NodeNumber* is small.

The channel assignment procedure in CA-OLSR is invoked on-demand, i.e. only when a channel conflict occurs after a topology change. Further, CA-OLSR seeks to assign distinct channels to active nodes in the two-hop neighborhood to minimize both primary and secondary collisions.

4.7 Formal Descriptions of Distributed Channel Assignment Protocols

In this section, I give a formal description of the proposed channel assignment protocols. Note that only procedures and steps that are relevant to channel assignment are presented.

A flow chart that describes the operation of CA-AODV is shown in Figure 4.6, while the operation of CA-OLSR is illustrated in Figure 4.7.

Figure 4.8 to Figure 4.11 describe the four main procedures that are common to CA-AODV, E2-CA-AODV, and E*k*-CA-AODV, namely sendRREQ(), recvRREQ(RREQ), sendRREP(), and recvRREP(RREP). Procedures sendRREQ() and recvRREQ(RREQ) are invoked in the route discovery phase, while sendRREP() and recvRREP() are invoked in the route reply phase.

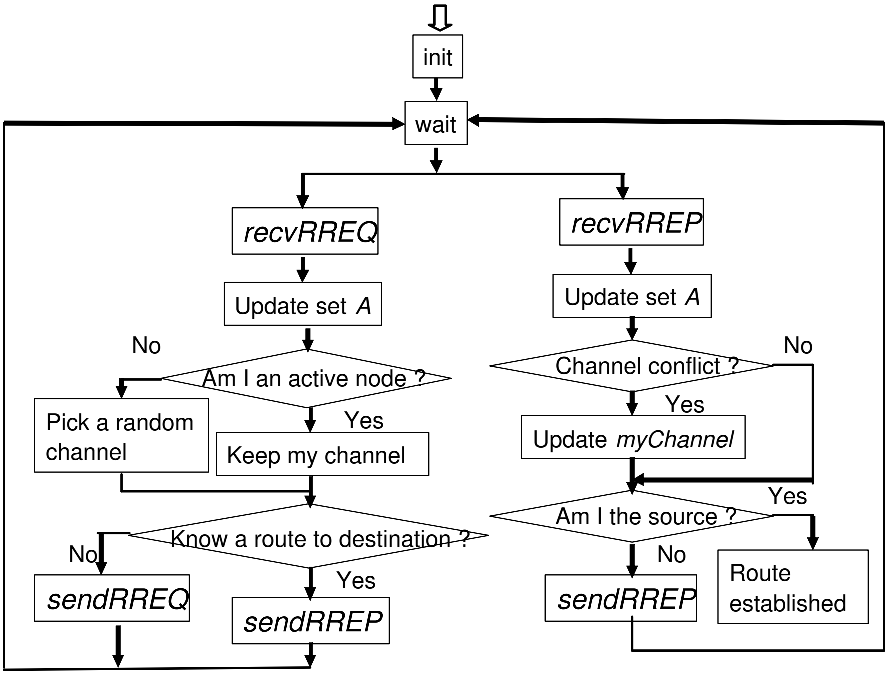


Figure 4.6: Flow chart of CA-AODV

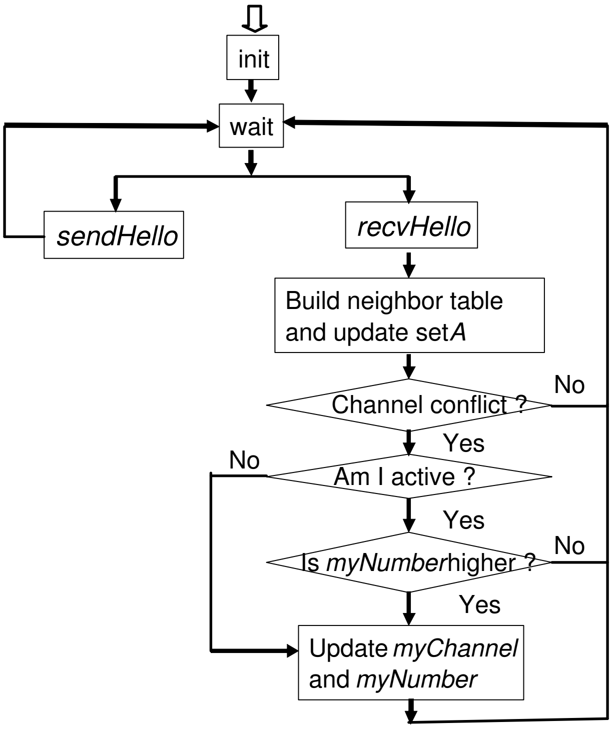


Figure 4.7: Flow chart of CA-OLSR

```
Procedure sendRREQ( )
  if myChannel == INVALID_CHANNEL
    myChannel = randomChannel ( A );
  endif
  for i=1 to k
    RREQ.neighborChannelIndex[ i ] = neighborChannel [i ] ;
  endfor
  Broadcast RREQ;
endprocedure
```

Figure 4.8: Procedure sendRREQ()

```
Procedure recvRREQ( RREQ )
  rt = getRouteInfo(RREQ);
  neighborChannel = getNeighborInfo( RREQ);
  if route is not in the routing table
    addRoute(rt);
  endif
  Update the available channel set A;
  if myChannel == INVALID_CHANNEL
    myChannel = randomChannel(A);
  endif
  if myIndex == destination || I know a route to destination
    sendRREP( );
  else
    sendRREQ( );
  endif
endprocedure
```

Figure 4.9: Procedure recvRREQ(RREQ)

Figure 4.12 and Figure 4.13 describe the sendChannelTaken() and recvChannelTaken() procedures respectively. These two procedures are only for the Ek-CA-AODV protocol.

The sendHello() and recvHello(HELLO) procedures for CA-OLSR are described in Figure 4.14 and Figure 4.15. In the recvHello(HELLO) procedure, the *getActiveNeighborInfo(HELLO)* function retrieves channel and *NodeNumber* information from `HELLO` messages received from neighboring nodes, while the *randomChannel(A)* function returns a channel index that is randomly chosen from the available channel set $\mathcal{A}$.

E2-CA-AODV utilizes three types of routing control messages, namely `RREQ`, `RREP`, and

```
Procedure sendRREP( )
    if myChannel != INVALID_CHANNEL
        RREP.myChannelIndex = myChannel;
    else
        RREP.myChannelIndex = INVALID_CHANNEL;
    endif
    for i=1 to k
        RREP.neighborChannelIndex [ i ] = neighborChannel[ i ] ;
    endfor
    Unicast RREP;
endprocedure
```

Figure 4.10: Procedure sendRREP()

```
Procedure recvRREP( RREP)
    neighborChannel = getNeighborInfo (RREP);
    Update the available channel set A;
    if Detect a channel conflict
        channelConflict = TRUE;
    endif
    if channelConflict == TRUE
        myChannel = randomChannel ( A);
    endif
    RREP.myChannelIndex = myChannel;
    for i=1 to k
        RREP.neighborChannelIndex [ i ] = neighborChannel[ i ];
    endfor
    Unicast RREP;
endprocedure
```

Figure 4.11: Procedure recvRREP(RREP)

```
Procedure sendChannelTaken( )
    ChannelTaken.channelIndex = myChannel;
    ChannelTaken.TTL = k;
    Broadcast ChannelTaken message;
endprocedure
```

Figure 4.12: Procedure sendChannelTaken()

```
Procedure recvChannelTaken(ChannelTaken)
  if myChannel == ChannelTaken.channelIndex
    ChannelConflict = TRUE;
  endif
```

```
Procedure sendHello()
  HELLO.myChannelIndex = myChannel;
  HELLO.NeighborNumber[0] = myNumber;
  for i=1 to k
    HELLO.NeighborchannelIndex [ i ] = NeighborChannel [ i ];
    HELLO.NeighborNumber [ i ] = NeighborNumber [ i ];
  endfor
  Broadcast HELLO
endprocedure
```

Figure 4.14: Procedure sendHello()

```
Procedure recvHello(HELLO)
  (neighborChannels, NodeNumber) = getActiveNeighborInfo (HELLO);
  Update the available channel set A;
  channelConflict = FALSE;
  if Detect a channel conflict
    if (activeNode == TRUE)
      if (activeMe == FALSE)
        channelConflict = TRUE;
      else if (NodeNumber <= myNumber)
        channelConflict = TRUE;
      endif
    else
      if (activeMe == FALSE & NodeNumber <= myNumber)
        channelConflict = TRUE;
      endif
    endif
  endif
  if channelConflict == TRUE
    myChannel = randomChannel(A);
    myNumber = randomNumber( );
  endif
endprocedure
```

Figure 4.15: Procedure recvHello(HELLO)

`HELLO` messages. Therefore, in addition to the sendRREQ(), recvRREQ(RREQ), sendRREP(), and recvRREP(RREP) procedures, E2-CA-AODV also has two extra procedures, sendHello() and recvHello(HELLO), which are similar to the procedures described in Figure 4.14 and Figure 4.15. In the recvHello(HELLO) procedure, the *getActiveNeighborInfo(HELLO)* function retrieves channel and *NodeNumber* information from `HELLO` messages received from neighboring nodes, while the *randomChannel(A)* function returns a channel index that is randomly chosen from the available channel set $\mathcal{A}$.

4.8 Analysis of the Protocols

In this section, I will first show that Ek-CA-AODV operates correctly and the number of channels required has an upper bound of $n \cdot (k+1)$, where n is the number of active routes that lie within the k-hop neighborhood of each other. Secondly, I show that CA-OLSR can assign distinct channels to any active node within a two-hop range. Finally, a numerical example is presented to show that Ek-CA-AODV can give a channel assignment similar to one provided by the centralized greedy algorithm.

4.8.1 Properties of CA-AODV and Ek-CA-AODV

First, the number of channels required by Ek-CA-AODV is shown to have an upper bound. The same upper bound can be applied to CA-AODV as well. Second, I show that the operation of Ek-CA-AODV is correct. The proof for CA-AODV is similar and will not be elaborated here.

Proposition 1. *To assign distinct channels to any node within a k-hop range, the number of channels required has an upper bound of $n \cdot (k+1)$, where n is the maximum number of active routes that lie within the k-hop range of any node in the network.*

Proof. This proposition can be proven by induction.

1. Base Case: If there is only one route in the network, it can be easily shown that the number of required distinct channels is $k + 1$.
2. Induction Step: Assume that when there are n active routes within a k-hop range, the required number of channels is $n \cdot (k + 1)$. If there are $n + 1$ active routes within a k-hop range, the $(n + 1)$-th route can be assigned $k + 1$ new channels that are different from any of the previous $n \cdot (k + 1)$ channels. The total number of channels needed for $n + 1$ active routes is then $(n + 1) \cdot (k + 1)$. Therefore, the proposed algorithms need at most $n \cdot (k + 1)$ distinct channels.

□

Note that this upper bound is achieved when there are no common nodes among active routes. If that is the case, channels are assigned to each active route independently and each route requires $k + 1$ channels. If active routes in the network have common nodes, the network requires fewer than $n \cdot (k + 1)$ channels.

Proposition 2. *After a new route has been found, each node along the new route is assigned a distinct channel among its k-hop neighbors, provided that the number of available channels is sufficient and W_t is large enough.*

Proof. Under the assumption that the channel assignment procedure is not disrupted by sudden failure or malfunction of nodes, the channel information carried by control messages will be consistent with the channel information saved at each node. Moreover, `ChannelTaken` messages can propagate to all nodes within a k-hop range, provided that W_t is large enough. Thus, after a new route has been found, each node on the route will have its k-hop active neighbors' channel information. Because a node randomly picks its own channel only from the available channel set, which does not contain any of its k-hop neighbors' channels, this node must have a channel that is distinct from any of its k-hop neighbors. □

Most of the existing algorithms fail to address the case when the number of available channels is less than the number of nodes in the neighborhood. However, the proposed algorithms take this case into consideration. If $k \leq 2$, a node with $\mathcal{A} = \emptyset$ should randomly pick a channel from $\mathcal{C}$. If $k \geq 3$, this node should randomly pick a channel from the channels that are taken by nodes two hops away. Thus, the algorithms attempts to ensure a minimum number of collisions, as well as reduce interference.

4.8.2 Correctness of CA-OLSR

Under the assumption that the number of available channels is sufficient and the channel information is updated more often than the topology changes, I show that CA-OLSR can assign distinct channels to any active node within a two-hop range.

For the purpose of this analysis only, CA-OLSR is assumed to operate under the following conditions.

1. The number of available channels is more than the number of active nodes within a two-hop range at any time.
2. All `HELLO` messages transmitted over a radio link are received correctly in the same hello interval.
3. Topology and channel changes can be conveyed to all nodes within a two-hop neighborhood before the next topology change. This could be achieved by choosing an appropriate HELLO interval according to the current mobility level.

Here, I assume that `HELLO` messages are sent frequently enough that a change in topology can be detected by nodes in the two-hop range within one or two hello intervals. This is also to assume that the frequency of topology changes is less than the topology update frequency. If the aforementioned assumptions do not hold for some situations, CA-OLSR can still function well, but its performance will likely not optimal.

Theorem 4.8.1. *Within a finite time after a topology change, CA-OLSR can assign distinct channels to any active nodes within a two-hop neighborhood.*

To prove the above statement, I need to show that the following statements are true.

1. CA-OLSR is dead-lock free and live-lock free.
2. Within a finite amount of time after a topology change, all nodes must have consistent and up-to-date topology and channel information about nodes within their two-hop neighborhoods.
3. If the information in the node is consistent and up to date, then CA-OLSR can assign channels correctly, such that no two active nodes in the two-hop neighborhood share the same data channel.

Lemma 1. *CA-OLSR is deadlock free and live-lock free. In addition, the channel updating procedure will take at most* $l-1$ *steps, where* $l \geq 2$ *is the number of active nodes that are involved in a channel conflict.*

Proof. The proof of the deadlock free property of CA-OLSR is straight forward. In CA-OLSR, each node in the network is assigned only one channel. However, for deadlocks to occur, a node must hold at least one resource, i.e. a channel, before it requests another resource. Since CA-OLSR assigns only one channel to each node, there is no deadlock.

Next, I prove that CA-OLSR is also live-lock free. In CA-OLSR, a node updates its data channel only if the following three conditions are satisfied at the same time.

1. There is a channel conflict between a node and at least one of its one-hop or two-hop neighboring nodes.
2. At least one neighboring node that has the same channel as the current node is active.
3. The node ID of the conflicting neighbor is less than that of the current node.

Note that channel conflicts that involve only inactive nodes do not affect network performance and, thus, are not resolved by CA-OLSR.

Three scenarios may occur where at least one node in the network needs to switch its data channel.

Case I: Assuming that there are exactly two active nodes that choose the same data channel, one node must have a node ID that is less than that of the other node because node IDs are unique within the network. Then, the node with lower node ID keeps its data channel while the node with higher node ID updates its own data channel to avoid the channel conflict. In this two-node case, no live-lock will occur.

Case II: Assuming that there are $l > 2$ active nodes that pick the same data channel, there must be a node that has the lowest node ID among the l active nodes. The node with the lowest node ID keeps its data channel, while each of the other $l - 1$ nodes pick a new data channel. Note that because the original data channel has been marked as unavailable at each of the $l - 1$ nodes, all of them pick a data channel that is different from the original one. In the worst-case situation where all $l - 1$ nodes pick the same data channel again, the node that has the lowest node ID among the $l - 1$ nodes can retain its data channel while the other $l - 2$ nodes randomly pick a new data channel from the available channel set $\mathcal{A}$. The available channel set $\mathcal{A}$ at each node is updated at each step. The channel update procedure continues until all l nodes have distinct data channels. In the worst case, this takes $l - 1$ steps. Because at every step the node that can retain its original channel is determined by its ID, which is unique in the network, no live-lock can occur either.

Case III: In the case that there are two or more different channel conflicts that involve a total of l active nodes, the same upper bound still holds. First, because each node is assigned only one channel at any time, different channel conflicts always involve different nodes. Thus, different channel conflicts can be resolved independently in parallel. The worst-case scenario occurs when at the second step, all remaining nodes pick the same data channel. Then, this situation degenerates into Case II.

As can be seen from the above three cases, no deadlock and live-lock can occur in CA-OLSR. In addition, the channel update procedure takes at most $l - 1$ steps, where l is the number of active nodes that have a channel conflict. ☐

Corollary 4.8.2. The value of the hello interval should be $3(l-1)$ times smaller than the topology change interval, where l is the maximum number of neighbors that can pick the same channel at the same time.

Proof. After updating its own data channel, a node v should send the updated topology information and channel information to its one-hop neighbors $N_1(v)$ in either the current hello interval, h_0, or the next hello interval, h_1. Note that if the change of channel happens after a node sends out a `HELLO` message, the change will be conveyed to its one-hop neighbors in the next hello interval. To get the upper bound, we assume that the updated `HELLO` message is always sent out in the second hello interval h_1. Upon receiving the updated `HELLO` message, one-hop neighbors $N_1(v)$ of the current node convey the updated channel information to the two-hop neighbors $N_2(v)$ by sending their `HELLO` messages in the next hello interval. Therefore, a channel update may take up to three hello intervals to reach all the nodes within a two-hop range.

The proof of Lemma 1 shows that it takes at most $l-1$ steps to resolve channel conflicts that involve l active nodes. To complete the channel update procedure, the updated channel information may need to propagate to all nodes in the two-hop neighborhood at each step. Since each step of the channel update procedure takes up to three hello intervals, the whole procedure takes up to $3(l-1)$ hello intervals. Because the channel update procedure has to be completed before the next topology change, the hello interval should be $3(l-1)$ times shorter than the topology change interval. ☐

Lemma 2. *Within a finite amount of time after a topology change, all nodes have consistent and up-to-date topology and channel information about nodes within their two-hop neighborhoods.*

Proof. Consistent and up-to-date topology and channel information means that a node knows the most recent topology change and all the recent channel changes among its one-hop and two-hop neighbors.

We know from Lemma 1 that CA-OLSR does not deadlock or live-lock. Specifically, the channel update procedure terminates after at most $l-1$ steps, where l is the maximum number of active nodes in the two-hop neighborhood that pick the same data channel. Because the number of nodes in the two-hop neighborhood is finite, l must be finite. Further, because we assume that the topology change is slow enough that topology update and channel update information can be conveyed to all nodes within the two-hop neighborhood before the next topology change, the next topology change will not interfere with the channel update procedure invoked by the current topology change. Therefore, within a finite time, the channel updating procedure terminates and all nodes eventually stop updating their available channel set $\mathcal{A}$ and their own data channels.

All `HELLO` messages transmitted over a radio link are assumed to be received correctly in the same hello interval. Thus, at most three hello intervals after the channel selection algorithm terminates, `HELLO` messages can propagate to any node's two-hop neighbors. Therefore, within a finite amount of time after a topology change, all nodes must have consistent and up-to-date topology and channel information about nodes within their two-hop neighborhood. □

Lemma 3. *After the channel update procedure terminates, no two active nodes in the two-hop neighborhood share the same data channel.*

Proof. The channel update procedure terminates when nodes do not make any new channel update. Since the topology and channel information in each node are consistent and up-to-date, as stated in Lemma 2, the channel update procedure terminates based on the correct topology and channel information. Because the channel update procedure terminates only when there is no channel conflict within any two-hop neighborhood, no two active nodes in the two-hop neighborhood share the same data channel. □

Based on the proven lemmas, I conclude that within a finite time after a topology change,

...	0	1	2	0	1	2	...
...	3	4	5	3	4	5	...
...	6	7	8	6	7	8	...
...	0	1	2	0	1	2	...

Figure 4.16: Channel assignment by the Greedy-AODV algorithm

...	0	1	2	0	1	2	...
...	5	4	3	5	4	3	...
...	8	7	6	8	7	6	...
...	0	1	2	0	1	2	...

Figure 4.17: Channel assignment by Ek-CA-AODV

CA-OLSR can assign distinct channels to any active nodes within a two-hop neighborhood. Thus, Theorem 4.8.1 is proven.

4.8.3 Comparison with a Centralized Algorithm

Through a numerical example, I show that Ek-CA-AODV can give channel assignment similar to the near-optimal Greedy-AODV algorithm. Greedy-AODV assigns channels to nodes on routes that are set up by the AODV routing algorithm in a greedy manner, assuming global knowledge of the network, routes, and channels.

In the numerical example, I use a random grid to represent a node distribution, which is a common practice in network simulations. A node is randomly placed in a grid, while within the grid, the node's position follows a random uniform distribution. To simplify the analysis, I also assume that, compared to the typical lifetime of active routes, the route setup time is negligible. Thus, no two routes are set up at exactly the same time.

Figure 4.16 illustrates the channel assignment by the centralized greedy algorithm. The first row represents the first established route; the second row represents the second established route, and so on. The number in each grid represents the channel index taken by the node in the grid. Here, k is assumed to be 2, meaning that the greedy algorithm seeks to assign distinctive channels to any node within a two-hop range. When the first route is being set up, Greedy-AODV assigns the lowest possible channel indices to the nodes on the first route, subject to the constraint that nodes within the two-hop range must be assigned distinct channels. The second route is set up after the first route. Since the lowest three channel indices have already been taken by the first route, which is within a two-hop range of the second route, Greedy-AODV assigns the lowest available channel indices to nodes on the second route. Three new channels are assigned to nodes on the third route. Since the fourth route is outside of the two-hop range of the first route, the channels assigned to nodes on the first route can be re-used. Note that this assignment is free of any conflicting channels within any node's two-hop range.

Figure 4.17 illustrates the channel assignment as made by Ek-CA-AODV. Note that the channels assigned to the nodes on the second and the third routes are in reverse order of the channels assigned by the Greedy-AODV algorithm. The reason is that the channel assignment is finalized during the Route Reply phase. Since each node picks the lowest channel index available and the channels are finalized from downstream nodes to upstream nodes, a downstream node will pick a lower channel index than its upstream neighbor. Note that this assignment, like that by Greedy-AODV, does not have any conflicting channels.

As can be seen from the above example, at least in some cases, Ek-CA-AODV can achieve the same performance as a near-optimal greedy algorithm.

4.9 Comparison with Other Channel Assignment Algorithms

4.9.1 Related Work

Hu's pioneering paper [32] examined distributed code assignment for CDMA packet radio networks, including wireless ad hoc networks. Using the assumption that each node has a neighbor table updated by a network-layer routing protocol, Hu's approach transforms the code assignment problem into a graph theory problem. Hu defines four code assignment schemes.

1. Common code assignment (CCA): All nodes in a network use a single common spreading code to transmit all packets;
2. Receiver-based code assignment (RCA): Each node is assigned a receiver code such that no two logical neighbors have the same code;
3. Transmitter-based code assignment (TCA): All neighboring nodes of a given node have different transmitter codes; and
4. Pair-wise code assignment (PCA): Each transmitter-receiver pair, i.e. an edge, is assigned a spreading code so that no two adjacent transmitter-receiver pairs or edges in the logical topology have the same code.

Both RCA and TCA are formulated as generalized graph coloring problems and PCA is formulated as a graph edge-coloring problem. Because both TCA and PCA problems are $\mathcal{NP}$-complete, fast heuristic algorithms are studied, including a greedy algorithm, a fan and chain re-coloring algorithm, and a sink-tree coloring algorithm. Even though the solutions proposed by Hu in [32] are sound, they have high time complexity and/or high communication overhead. Moreover, the schemes do not consider the case when the number of codes is limited and perfect assignment is not possible. Hu's scheme also assumes that each node has knowledge of its neighbors through a routing table. While this is applicable in most cases, assigning channels after the routing protocol converges could incur excessive delay and control overhead.

Garcia-Luna-Aceves and Raju [23] describe a distributed code assignment scheme (CAS) that works in a mobile ad hoc network. CAS assigns distinct channels, or codes, to a node and its two-hop neighbors. Each node sends out code assignment messages (CAM) that propagate up to one hop away from the node. CAMs are sent in three conditions: 1) when a new node comes up, 2) when a node detects a change of code by any of its one-hop neighbors, and 3) when a node finds that one of its one-hop neighbors is no longer active. After receiving the CAM, each neighbor needs to acknowledge the message individually. Each CAM contains: 1) the address and the code of the node that is sending out the CAM, 2) the addresses and the codes of the node's one-hop neighbors, 3) acknowledgements to earlier CAMs, and 4) a response list of zero or more nodes which need to send an ACK for this CAM. If the number of codes available for assignment is at least $d(d-1)+2$, where d is the maximum degree, i.e., the maximum number of neighbors for any node, it is shown that there will be no interference after the algorithm converges. However, this algorithm incurs high communication and computation overhead and has high time and storage complexity. In addition, the algorithm does not specify what to do if the number of available channels is less than $d(d-1)+2$.

Wu, *et al.* [76] propose a dynamic channel assignment (DCA) protocol that can be applied to both FDMA and CDMA systems. Using the assumption that there is one control channel and a fixed number of data channels available in a network, DCA assigns channels in an "on-demand" manner, which means channels are assigned only to mobile hosts that intend to transmit. Once a host completes its transmission, the channel is released. Because a channel needs to be assigned to a host dynamically before each packet transmission, DCA is invoked frequently, which leads to high computational overhead. Moreover, because DCA operates at the MAC layer, which only has information of nodes within one hop, it is difficult to extend the protocol to consider nodes more than one hop away.

There are several recent proposals for routing protocols that are suitable for multi-hop multi-channel wireless mesh networks [59, 64, 46, 69]. The approach taken by most of these proposals is to combine routing with intelligent multi-channel assignment, such that channel utilization is maximized and the system performance can be substantially improved.

Raniwala and Chiueh [59] propose to utilize local traffic load information to dynamically assign channels and to route packets according to routing metrics such as path capacity. Path capacity is defined as the minimum residual bandwidth of a path. Two channel assignment issues are addressed by Raniwala and Chiueh [59]: neighbor-interface binding and interface-channel assignment. Neighbor-interface binding decides how to bind a node's interfaces to neighboring nodes because two neighboring nodes need to be on the same channel to communicate. Interface-channel assignment assigns radio channels to nodes' interfaces without global coordination, such that traffic loads are balanced among channels. Raniwala and Chiueh [59] assume a static network, where neighboring nodes do not change through the duration of the network. However, traffic patterns and, thus, channel loads can evolve over time. Therefore, the neighbor-interface binding is not changed once it is completed, whereas the interface-to-channel mapping is adjusted periodically. By combing routing with intelligent channel assignment, network good-put is substantially improved while the delay is reduced. Simulation results show that by deploying just two network interface cards (NICs) per node, the distributed channel assignment/routing algorithms can achieve a factor of 6 to 7 throughput improvement compared to the conventional single-channel scheme. Similar problems have also been investigated in [64, 46]. While So and Vaidya [64] study routing and channel assignment with single-NIC devices, Kyasanur and Vaidya [46] propose a routing protocol that can work in general multi-radio mobile ad hoc networks.

However, because the prior work described above emphasizes the routing protocols, the effectiveness of the channel assignment schemes has not been studied [59, 64, 46]. Some channel assignment protocols have very high time complexity, i.e. $O(Kn^3 \log m + m^2)$ [69], where n is the total number of nodes in the network, m is the total number of radio connections in the network, and K is the minimum number of neighbors for any node. Some proposals utilize simple channel assignment schemes, for example Kyasanur and Vaidya [46] use a scheme that is similar to random channel assignment.

Since it has been proven that the distributed channel assignment problem is $\mathcal{NP}$-complete and, thus, computationally intractable, many heuristic algorithms have been proposed. However, most of them require more channels than necessary and have high complexity [6, 23, 32, 76, 69]. For instance, they often assign channels to all nodes in the network even though some nodes may not be active. These algorithms also have high communication and computational overhead. In addition, they ignore harmful interference that may corrupt packets and cause retransmissions.

4.9.2 Comparison of Complexity

Table 4.1 compares several heuristic algorithms in terms of their communication, computation, and storage complexity.

For CA-AODV and E2-CA-AODV, the control messages are carried solely by routing messages. In addition, channel assignments for each active route are saved along with each routing entry. Therefore, there is very little communication overhead and storage overhead. Ek-CA-AODV introduces a `ChannelTaken` message that is broadcast by nodes on n active routes to nodes within a k-hop neighborhood. Thus, tEk-CA-AODV introduces a ChannelTaken message that is broadcast by active nodes to nodes within a k-hop neighborhood. Thus, the communication complexity of Ek-CA-AODV is $O(|V_t| \cdot |V_k|)$, where $|V_t|$ is the number of transmitters and $|V_k|$ is the number of active nodes in the k-hop neighborhood. CA-OLSR also utilizes routing control messages exclusively to carry channel information. In addition, channel assignments of neighbors are saved along with each entry in the neighbor table. Therefore, there is very little communication overhead and storage overhead. The random scheme does not require a particular number of channels to operate because it allows each node to randomly pick a channel. Additionally, the random scheme has the minimum complexity among all channel assignment schemes. Because the centralized greedy algorithm is not a distributed algorithm, the communication complexity metric does not apply to it.

In Table 4.1, d is the maximum number of neighbors for any node, d_a is the maximum number of active neighbors for any node, n is the total number of active routes within k-hop range of each other, and k is a constant associated with the interference range. Note that both [32] and [23] only consider two-hop neighbors.

Table 4.1: Comparison of Proposed Channel Assignment Algorithms to Existing Algorithms

Protocols	*Number of Channels*	*Communication Complexity*	*Computational Complexity*	*Storage Complexity*
Centralized greedy algorithm [32]	$d(d-1)+1$	N/A	$d^2 \cdot \lvert V \rvert$	$d^2 \cdot \lvert V \rvert$
Distributed channel assignment [23]	$d(d-1)+2$	$d^2 \cdot \lvert V \rvert$	d^2	d^2
CA-AODV	$k+1$	$O(k)$	$O(k)$	$O(k)$
E2-CA-AODV	$d_a(d_a-1)+1$	$O(d_a)$	$O(d_a^2)$	$O(d_a)$
Ek-CA-AODV	$n(k+1)$	$O(\lvert V_t \rvert \cdot \lvert V_k \rvert)$	$O(\lvert V_k \rvert)$	$O(\lvert V_k \rvert)$
CA-OLSR	$d_a(d_a-1)+1$	$O(1)$	$O(d_a^2)$	$O(1)$
Random scheme	N/A	$O(1)$	$O(1)$	$O(1)$

4.10 Summary

Three design principles for distributed channel assignment are discussed in this chapter. The first principle is a *cross-layer* design principle, the second principle is an *on-demand* channel assignment principle, and the third principle is an *interference mitigation* principle.

Two example implementations of these design principles are introduced; CA-AODV is based on a reactive routing protocol, while CA-OLSR is based on a proactive routing protocol. CA-AODV also has two extensions, E2-CA-AODV and Ek-CA-AODV. I prove the correctness of the proposed protocols and derive an upper bound on the number of channels required to resolve collisions as well as to mitigate mutual interference. These channel assignment protocols exhibit significantly lower communication, computation, and storage complexity than existing channel assignment schemes. In addition, they require fewer channels than many existing channel assignment algorithms.

Chapter 5

A New Multi-Channel MAC (MC-MAC) Protocol

In this chapter, I present a new multi-channel MAC (MC-MAC) protocol that works with the proposed channel assignment protocols.

Traditionally, a multi-channel MAC usually consists of two major management components: channel assignment and medium access control. Channel assignment determines which channel is to be used by which node, while medium access control is responsible for coordinating the access of a particular channel, resolving contention and collisions on that channel. The channel assignment component is designed to utilize as many channels as possible within a neighborhood and to keep channel conflicts to a minimum. Meanwhile, the medium access control component guarantees that even when two neighboring nodes select the same channel, contention and collisions on the channel are effectively controlled and resolved.

The goals of multi-channel MAC design are twofold: 1) to achieve a high channel throughput, and 2) to reduce delay. These goals are achieved by the channel assignment component that seeks to maximize spatial re-use of multiple channels and by the medium access control component that tries to minimize contention and collisions on each individual channel.

In addition to the two main design goals, there are secondary design objectives such as fairness, scalability, energy efficiency, and support for quality of service (QoS). To date, most MAC protocols only consider the two main design goals. The secondary design goal of MC-MAC, proposed here, is to provide fair medium access in the network. As far as I know, none of the existing multi-channel MAC protocols specifically addresses this or other secondary design goals.

Because the channel assignment function is performed by extended ad hoc routing protocols, such as CA-AODV or CAOLSR, the MC-MAC protocol only needs to manage medium access control on multiple data channels. In addition, MC-MAC is essentially compatible with the single-channel IEEE 802.11 MAC due to the following reasons.

- Like the IEEE 802.11 MAC, MC-MAC is based on CSMA/CA.
- MC-MAC requires only one half-duplex transceiver per host and does not require synchronization among nodes. This is exactly the same system requirement for an IEEE 802.11 node. As far as we know, the MC-MAC protocol has the minimum system requirement among all multi-channel MAC protocols proposed to date.
- Like the IEEE 802.11 MAC, MC-MAC only needs to perform medium access control, whereas most other multi-channel MAC protocols must also perform channel assignment. In our proposed protocols, channel assignment is performed by the routing layer instead of the MAC layer.

While the IEEE 802.11 MAC protocol is quite robust, it can be overly conservative. For instance, the well known exposed node problem can lead to unnecessary transmission deferrals and, thus, lead to low end-to-end throughput and high packet-delivery latency [8]. In addition, the binary exponential backoff mechanism is shown to be unfair and sometimes inefficient, especially when there are many active nodes [9]. As stated in Chapter 2, the

IEEE 802.11 MAC protocol treats all transmission failures or delays as collisions, whereas many of them are not necessarily collisions.

Compared with the IEEE 802.11 MAC, MC-MAC has two unique design features.

1. MC-MAC does not utilize a binary exponential backoff (BEB) scheme as does the IEEE 802.11 MAC. Instead, each node has the same pre-determined contention window size to ensure maximum fairness in the network. MC-MAC assumes that there is one dedicated control channel and up to N data channels in the network. If the number of available channels is sufficient, collisions will occur only on the common control channel. Further, in MC-MAC, contention on the common control channel is much less severe than that in the IEEE 802.11 MAC, mainly because control packets are much shorter than ordinary data packets. Therefore, the MC-MAC protocol can utilize channels more aggressively than the IEEE 802.11 MAC.
2. MC-MAC utilizes a modified virtual carrier sensing mechanism to mitigate the exposed node problem and to improve performance. When a node hears RTS and CTS messages that indicate use of a different data channel than it is using, it will defer only to the end of the CTS reception. If the RTS and CTS messages indicate that the future data transmission will use the same channel as its own channel, a node will defer until the end of the data transmission only when it hears the CTS message or the transmission of a data packet.

In the next section, I give a detailed description of the proposed MC-MAC protocol.

5.1 Description of the MC-MAC Protocol

The multi-channel MAC (MC-MAC) protocol is designed to allow simultaneous data transmissions on different data channels. It is assumed that there is one dedicated control channel and up to N data channels in the network. Each data channel is equivalent and has the same ban -

gle s

sim

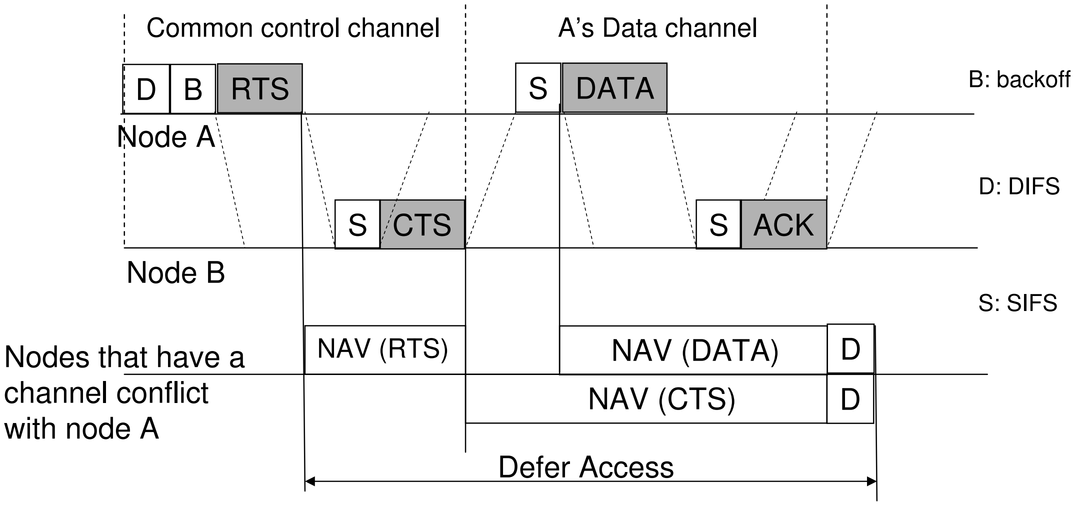


Figure 5.1: Four-way handshake procedure of MC-MAC

MC-MAC is a transmitter-based protocol. Nodes are assigned channels by the combined channel assignment and routing protocols. RTS and CTS messages are sent on the control channel, while the data packets and ACK messages are sent on the transmitter's assigned data channel. When a node is ready to transmit, it will first convey its assigned data channel to the destination node. As shown in Figure 5.1, when a sender node A intends to transmit, it first broadcasts a RTS message carrying its own data channel index, c_A. Upon receiving the RTS message, the destination node B sends back a CTS message carrying c_A and switches its receiving channel to c_A. After node A receives a CTS message, it switches to the designated data channel and starts data transmission. If an ACK is received after the completion of the data packet transmission or if no ACK is received before a timeout occurs, the sender will switch back to the control channel. Similarly, after sending the ACK message on the data channel, the receiving node also switches back to the control channel. In general, when a node is not engaged in transmitting or receiving on a data channel, it should always listen on the control channel.

To ensure maximum fairness and to utilize channels more efficiently, MC-MAC does not implement a BEB scheme. Thus, failed retransmissions of an RTS message or failed reception of an ACK message will not increment the contention window. There are three reasons for this design choice. First, the original IEEE 802.11 MAC protocol invokes BEB upon all failed on-time receptions. However, there are many cases in which an increase in the contention window size is not necessary, especially when there are multiple channels available. In fact, as has been pointed out by Bharghavan, *et al.* [8], the BEB mechanism that provides fair and efficient medium access in wired network becomes the cause of unfairness in wireless networks. Secondly, in MC-MAC, if there are sufficient data channels available, contention and collisions occur only on the common control channel. Thirdly, control packets are often much smaller than typical data packets. Because contention on the control channel is less severe in MC-MAC than in the IEEE 802.11 MAC which uses a single channel, MC-MAC can utilize available medium more aggressively. Due to these three reasons, MC-MAC performs better without a binary exponential backoff scheme.

In the IEEE 802.11 MAC protocol, BEB is invoked whenever an assumed collision happens. However, in many cases, the contention window size is unnecessarily increased.

- In the four-way handshake, if a sender does not receive a CTS in response to its RTS before a timeout, it will assume that a collision has occurred and start BEB. This is irrespective of the destination node. For example, the destination may defer the transmission of the CTS message because it detects a transmission in its vicinity.

  In a multi-channel MAC protocol, this type of scenario may occur more often. For instance, the destination node might be busy transmitting or receiving on another channel and cannot receive RTS messages on the control channel. This type of terminal is called “deaf terminal.” If each node is only equipped with one half-duplex transceiver and nodes are not synchronized when they switch channels, it is difficult to avoid the “deaf terminal” problem. A BEB scheme will only exacerbate the problem. Since “deaf terminals” are not contending with the current node, there is no need to increase the contention window.

- A packet transmitted by a sender may be corrupted due to noise or accumulated interference generated from nodes that are more than one-hop away from the receiver. The corrupted packet is mistaken as being caused by a collision by both the sender and the receiver.

- When the number of available channels is insufficient for assigning distinct channels to neighboring nodes, two or more nodes may pick the same data channel. When two of these nodes do not know that there is a channel conflict and they try to transmit at the same time, collisions may occur. A similar scenario has also been observed by So and Vaidya [63], where it is called the "multi-channel hidden terminal" problem. Note that MC-MAC allows nodes to exchange channel information on the common control channel. Therefore, such a "multi-channel hidden terminal" problem may be avoided.

- Due to a *capture effect*, two simultaneous data transmissions on the same channel may not result in a collision [66]. Instead, the node that is closer to the destination may be able to successfully transmit its data packet. Because there is no collision, there is no need to increase the contention window.

If all of the above scenarios are treated as collisions, contention windows will be unnecessarily increased. For MC-MAC, because all data traffic is spread over independent data channels and the control packets are much shorter than data packets, collisions on the control and data channels are much less likely, compared to the IEEE 802.11 MAC protocol. Therefore, MC-MAC performs better without the binary exponential backoff procedure.

MC-MAC also incorporates a modified virtual carrier sensing (VCS) mechanism, called the intelligent virtual carrier sensing (IVCS) scheme as explained later, which is different from the scheme used by the IEEE 802.11 MAC. In the IEEE 802.11 MAC, nodes that overhear either CTS or RTS messages should defer until the end of the data transmission regardless of whether or not a new packet transmission will interfere with the ongoing packet

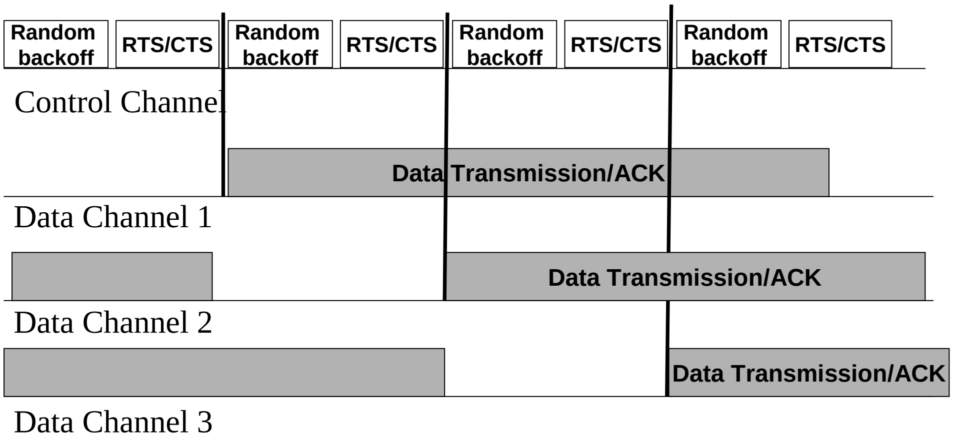


Figure 5.2: Efficient pipelining by MC-MAC

transmission.

In MC-MAC, all nodes that overhear the RTS and CTS messages first examine their own channel indices. Nodes that do not use the same channel as the on-going data transmission defer only for the duration of the control packet transmission. If a node is assigned the same data channel as the one used by the current transmitting node, c_A, two situations may happen, as illustrated in Figure 5.1.

- If a node overhears a CTS message, it should defer from using data channel c_A until the end of the data transmission to avoid causing a collision at the receiver.
- A node that overhears only an RTS message, but not a CTS message, should first defer from using the control channel only for the duration of the control packet transmission. Then, it performs carrier sensing on its own data channel c_A. If the carrier is busy, which means that the transmitting node has successfully acquired the medium, the node should defer for the duration of the data packet transmission. However, if the carrier is not busy, the node need not defer any longer.

The purpose of utilizing the modified VCS scheme is two fold: 1) to mitigate the exposed node problem and 2) to achieve efficient "pipelining." When there are not sufficient channels available, two or more nodes in the same neighborhood may pick the same data channel.

If one of them is transmitting, other nodes that pick the same data channel become the exposed nodes. Note that nodes that do not have channel conflict do not become exposed nodes because they can essentially have their own data channels to transmit on. Because this modified VCS scheme allows nodes to intelligently identify exposed nodes and tries to mitigate the exposed node problem as much as possible, it is called the intelligent virtual carrier sensing (IVCS) scheme. Because MC-MAC transmits control packets on the common control channel and data packets on non-overlapping orthogonal data channels, the transmissions on the control channel and on different data channels can occur in parallel. This type of parallelism is sometimes called pipelining [79]. Figure 5.2 illustrates the pipelining effect achieved by MC-MAC. When nodes have different data channels from the on-going data transmissions, they need to defer only for the duration of control packet transmission. Since the size of typical data packets is usually much larger than that of control packets, many data transmissions can occur in parallel on different data channels. Therefore, by utilizing the IVCS scheme, MC-MAC can achieve efficient pipelining.

5.2 Related Work

There are many prior studies that examine the benefits of utilizing multiple channels. Most current multi-channel MAC protocols have two functionalities: channel assignment and medium access control. Based on how each function is designed and hardware requirements, multi-channel MAC protocols can be classified into different categories.

Based on the way that a channel is selected, existing approaches can be classified into two different categories: 1) handshake-based channel selection or 2) contention-based channel negotiation. Depending on how medium access control is performed, existing protocols fall into one of the following two categories: 1) CSMA/CA based random access approach or 2) channel hopping. Based on hardware requirements, schemes fall into three categories: 1) multiple transceivers are required, 2) one transmitter and multiple receivers are required, or 3) one transceiver is required. A transceiver is also called a network interface card (NIC).

There is no general rule as to which scheme is better than another. Simpler schemes with less hardware requirements are often compatible or nearly compatible with the IEEE 802.11 standard and are easier to implement, whereas complex schemes with greater hardware requirements often yield better performance.

Existing work can be classified based on other criteria as well. For instance, some protocols use a common control channel for all nodes, while others do not. The purpose of using a common control channel is to transmit control packets and assign data channels to mobile nodes. However, if the number of available data channels is larger than a certain value, the control channel may saturate.

Here we discuss categorization based on the three criteria stated above. First, based on the way that a channel is selected, existing approaches can be classified into the following two categories.

- *Handshake-based channel selection*: Both the dynamic channel assignment (DCA) [76] and the multi-channel CSMA MAC protocol [35] utilize handshake for channel selection. Like the IEEE 802.11 standard, the handshake is realized by sending control messages between senders and receivers. Handshake-based channel selection is widely used and quite flexible. However, for DCA and the multi-channel CSMA MAC, the negotiation of the channel to use is on a per-packet basis, which may result in high control overhead. In contrast, our proposed CA-OLSR scheme is invoked only when there is a channel conflict that needs to be resolved.

- *Contention-based channel negotiation*: Multi-channel MAC (MMAC) divides time into beacon intervals [63]. A small window is placed at the beginning of each beacon interval. During this window, all nodes will transmit and receive on a common control channel and contend to select the desired data channel to use. Once a sender and a receiver successfully agree on a data channel, both of them will switch to the data channel for transmission at the end of the contention window. However, broadcasting is not supported in MMAC, which poses challenges for ad hoc routing protocols that rely on broadcasting for route queries. In addition, MMAC required nodes in the network switch to a common control channel at the same time. Thus, all nodes in the network need to be synchronized, which may be hard to achieve in an ad hoc network.

According to the medium access control techniques utilized, multi-channel MAC protocols fall into one of the following two categories.

- *CSMA/CA based random access*: Many of the multi-channel MAC protocols follow this approach because carrier sensing and handshake mechanisms are still proven ways to avoid contentions and collisions on wireless channels. Examples include DCA [76], MMAC [63], and a multi-channel CSMA MAC protocol [35]. Our proposed MC-MAC protocol falls into this category.
- *Channel hopping*: Multi-channel MAC protocols that are based on logical channel hopping do not require carrier sensing and do not assign channels to individual nodes. Two such examples are receiver-initiated channel-hopping with dual polling (RICH-DP) [72] and slotted seeded channel hopping (SSCH) [5]. In RICH-DP, all nodes in the network hop from one channel to another according to the same pre-defined hopping sequence. Two nodes can start transmitting when one node has a packet for the other. They stay on the same channel until the data transmission is finished. Afterwards, they follow the same hopping pattern as the rest of the network. RICH-DP requires clock synchronization among nodes. SSCH assumes that each node can calculate and update its channel hopping sequence based on an initial channel index and a seed [5]. Nodes then switch channels from one time slot to the next according to the channel hopping sequence. In addition, nodes can learn each other's hopping schedules by broadcasting their channel schedules. When two nodes have overlapping channels at some time instances, they can start to transmit to each other. SSCH requires no dedicated control channel, but needs clock synchronization among nodes.

According to the number of required transceivers or NICs, existing multi-channel MAC protocols can be divided into three categories, as described below.

- *Multiple transceivers*: Some MAC protocols require more than one transceiver to take advantage of multiple channels. For example, DCA uses one transceiver to monitor a control channel and a second transceiver to transmit and receive on a data channel. With added hardware complexity, DCA can utilize multiple channels with moderate control overhead.
- *One transmitter and multiple receivers*: Jain, *et al.* assume that a node can receive packets on all channels simultaneously [35]. Thus, each node in a network must be equipped with one transmitter and multiple receivers.
- *One single half-duplex transceiver*: MMAC [63], MC-MAC, RICH-DP [72], and SSCH [5] all require that each node has only a single half-duplex transceiver. However, MMAC, RICH-DP, and SSCH require node synchronization to some extent in the network.

The multi-channel MAC schemes that require only a single transceiver can yield at least a factor two of improvement over single-channel MAC protocols [5, 63]. My simulation results show an improvement up to a factor of four. The decrease on packet delay is even greater due to the significant reduction of the number of collisions and back-offs. However, because there is only one NIC available, a node can only operate on one channel at a time, which may result in inefficient channel utilization. A multi-channel MAC scheme that utilizes multiple NICs may yield better performance at the cost of higher hardware complexity. Increasing the number of transceivers per node may also cause excessive energy drain and consequent reduction of battery life in mobile devices. This is a fundamental tradeoff between hardware complexity and system performance. Because channel assignment is combined with a routing protocol in our work, the MC-MAC protocol only needs to implement the medium access control function, which greatly reduces the hardware complexity. Table 5.1 summarizes some important features of MC-MAC in comparison to existing multi-channel MAC protocols.

With the proposed MC-MAC protocol, nodes need only a single half-duplex transceiver,

Table 5.1: Comparison of MC-MAC to Existing Multi-Channel MAC Protocols

Protocols	*Medium Access*	*Channel Selection*	*Hardware Requirement*	*Synchro. Required*
DCA [76]	CSMA/CA	Per packet	2 transceivers	No
MMAC [63]	CSMA/CA	Per beacon interval	1 transceiver	Yes
Multi-channel CSMA [35]	CSMA/CA	Per packet	1 transmitter multiple receivers	No
RICH-DP [72]	Channel hopping	Hopping sequence	1 transceiver	Yes
SSCH [5]	Channel hopping	Hopping sequence	1 transceiver	Yes
MC-MAC	CSMA/CA	Per route change	1 transceiver	No

like MMAC and SSCH. However, both MMAC and SSCH require clock synchronization of all nodes, which is difficult in an ad hoc network. In contrast, the MC-MAC protocol does not require nodes switch channels at the same time. Thus, node synchronization is not required in the network. Thus, MC-MAC has the minimum requirement among all existing multi-channel MAC protocols.

5.3 Summary

A multi-channel MAC (MC-MAC) protocol is presented in this chapter. MC-MAC has two design objectives: maximizing throughput and providing fairness to all nodes in the network. To the best of my knowledge, MC-MAC is the first multi-channel MAC protocol that explicitly takes fairness into consideration by utilizing the same fixed contention window size for all nodes in the network.

Compared with the IEEE 802.11 MAC, MC-MAC has two unique design features. First, MC-MAC does not utilize a binary exponential backoff scheme. This is to ensure maximum fairness to all nodes. Secondly, MC-MAC implements an intelligent virtual carrier sense scheme to intelligently identify exposed nodes and to mitigate the exposed node problem. In addition, MC-MAC is compatible with the IEEE 802.11 MAC protocol and only requires one half-duplex transceiver per host, which is the minimum system requirement among all existing multi-channel MAC protocols.

Chapter 6

Simulation Study and Capacity Analysis

Using results from the ns2 simulator, the performance of distributed channel assignment protocols, such as CA-AODV and Ek-CA-AODV, is examined in Section 6.1. Then, the capacity of multi-channel wireless networks is analyzed in Section 6.2. Simulation results are also presented to illustrate the capacity improvement of distributed channel assignment combined with MC-MAC compared to the single-channel IEEE 802.11 MAC. Various simulation scenarios are summarized in Table 6.1. Because Ek-CA-AODV is suitable for static wireless networks, the performance of the Ek-CA-AODV protocol is studied in four static wireless networks, namely a one-hop network, a chained network, a square grid network, and a non-square grid network. The performance of E2-CA-AODV and CA-OLSR is studied in several mobile ad hoc network scenarios, as summarized in Table 6.1.

Table 6.1: Simulation Scenarios

Simulation Scenario	*Network*	*Simulated Protocols*	*Section Number*
Preliminary simulation	static, 800 m by 800 m	CA-AODV and Ek-CA-AODV	6.1
One-hop network	static, 120 m by 120 m	Ek-CA-AODV	6.2.1
Chained network	static	Ek-CA-AODV	6.2.2
Square grid network	static	Ek-CA-AODV	6.2.3
Non-square grid network	static	Ek-CA-AODV	6.2.3
Dense mobile ad hoc network High mobility	800 m by 800 m 18 m/s to 19 m/s pause time: 5 s	E2-CA-AODV	6.2.4
Dense mobile ad hoc network Low mobility	800 m by 800 m 4 m/s to 5 m/s pause time: 5 s	E2-CA-AODV	6.2.4
Sparse mobile ad hoc network High mobility	1600 m by 1600 m 18 m/s to 19 m/s pause time: 5 s	E2-CA-AODV	6.2.4
Sparse mobile ad hoc network Low mobility	1600 m by 1600 m 4 m/s to 5 m/s pause time: 5 s	E2-CA-AODV	6.2.4
Dense mobile ad hoc network High mobility	800 m by 800 m 18 m/s to 19 m/s pause time: 5 s	CA-OLSR	6.2.4
Dense mobile ad hoc network Low mobility	800 m by 800 m 4 m/s to 5 m/s pause time: 5 s	CA-OLSR	6.2.4
Sparse mobile ad hoc network High mobility	1600 m by 1600 m 18 m/s to 19 m/s pause time: 5 s	CA-OLSR	6.2.4
Sparse mobile ad hoc network Low mobility	1600 m by 1600 m 4 m/s to 5 m/s pause time: 5 s	CA-OLSR	6.2.4

6.1 Preliminary Performance Evaluation of CA-AODV and Ek-CA-AODV

To evaluate the effectiveness of distributed channel assignment protocols, simulations are performed using 25 random scenarios with distinct source-destination pairs. Each random scenario has 64 wireless nodes that are randomly distributed over an 800 m by 800 m area. The two-ray ground path loss model [74] is used and the radio range is assumed to be 180 m. To compare with existing channel assignment schemes, static topologies are assumed in simulations. The total simulation period is 700 seconds that include a 200-second warm up period and 500 seconds of effective simulation time. I measured the average number of conflicting nodes within the interfering range as well as the average accumulated interference levels in the network. The number of conflicting nodes is defined in Equation 3.1 as the number of nodes that share the same data channel. The average accumulated interference level is defined in Equation 3.2. The values of the simulation parameters used in the preliminary performance evaluation are summarized in Table 6.2.

Table 6.2: Simulation Parameters Used in the Preliminary Performance Evaluation

Simulation Parameter	*Value*
Number of nodes	64
Network size	800 m by 800 m
Radio range	180 m
Interference range	360 m
Physical channel bandwidth	2 Mbps
Path loss model	Two-ray ground
Warm-up period	200 seconds
Effective simulation time	500 seconds
Mobility	No

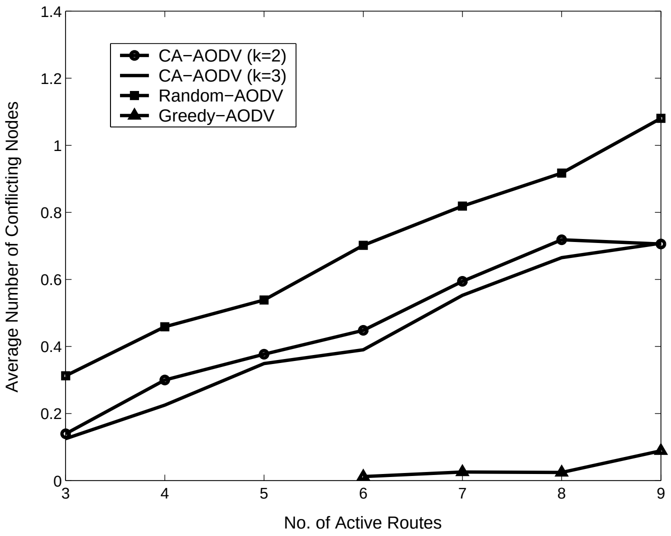


Figure 6.1: Average number of conflicting nodes (CA-AODV)

Figure 6.1 indicates that the number of conflicting nodes increases with an increase in the number of active routes. The x-axis represents the number of active routes and the y-axis represents the average number of conflicting nodes within the interference range. Random-AODV algorithm assigns channels randomly to nodes on active routes. Random-AODV has the least complexity and provides an upper bound on the number of conflicting nodes. CA-AODV has complexity similar to Random-AODV, but performs significantly better than Random-AODV. Greedy-AODV algorithm is a centralized near-optimal algorithm that seeks to assign distinct channels to any active node within the interference range. Note that Greedy-AODV can achieve zero channel conflicts when there are a small to medium number of active routes. The performance of CA-AODV improves slightly when k is increased from 2 to 3.

Assuming the worst-case scenario where all active transmitters transmit simultaneously, the accumulated interference level at each receiver can be measured, as shown in Figure 6.2. The x-axis represents the number of active routes and the y-axis represents average accumulated interference level in dB at each receiver. Figure 6.2 shows that the accumulated interference levels generated by Random-AODV range from -84 dB to -73 dB, which are

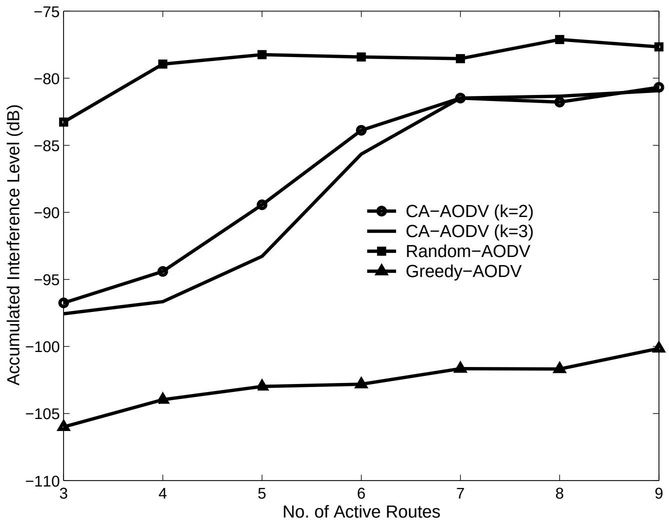


Figure 6.2: Accumulated interference level at each receiver (CA-AODV)

higher than the minimum received signal level of -88 dB for the 180 m radio range. When the number of active routes is small, CA-AODV works well and the performance gap between CA-AODV and Greedy-AODV is relatively small. However, the performance of CA-AODV degrades sharply when the number of active routes increases. The reason for this is that CA-AODV assigns channels to each active route independently through only AODV control messages. Thus, channel conflicts occur more frequently when the number of active routes increases. Since it is assumed that nodes use omnidirectional antennas, channel conflicts among neighboring routes will cause mutual interference.

Figure 6.3 and Figure 6.4 illustrate the performance gain of Ek-CA-AODV over CA-AODV. Since `ChannelTaken` messages convey channel information among different routes, the performance of Ek-CA-AODV degrades gracefully with an increase in the number of active routes. The performance of Ek-CA-AODV is always better than that of CA-AODV. Moreover, the performance gap between Ek-CA-AODV and CA-AODV increases when the number of active routes increases. This is because CA-AODV assigns channels to each active route independently using only AODV control messages. Thus, channel conflicts occur more frequently when the number of active routes increases. In contrast, Ek-CA-AODV can significantly reduce the number of channel conflicts among different routes by

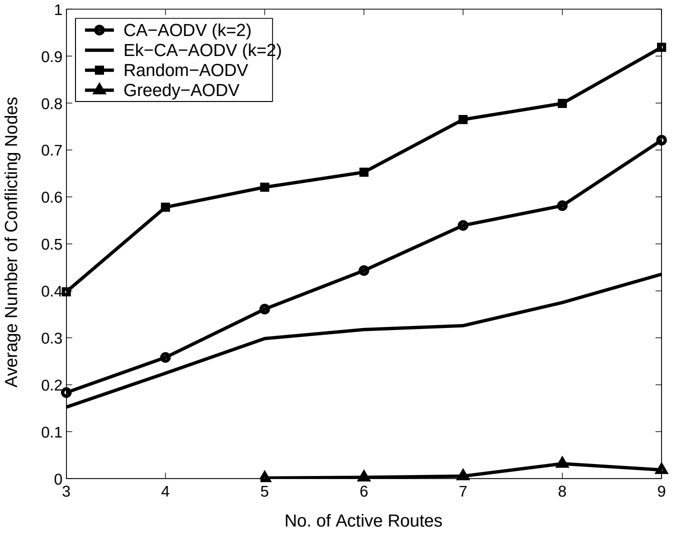


Figure 6.3: Average number of conflicting nodes (CA-AODV and Ek-CA-AODV)

using `ChannelTaken` messages. Thus, the performance of Ek-CA-AODV degrades more gradually.

Figure 6.5 indicates that when the neighborhood size k increases, the performance of Ek-CA-AODV improves. Note that when $k = 3$, the performance gap between Ek-CA-AODV and Greedy-AODV becomes very small. I conjecture that if neighborhood size k and wait time W_t are set to appropriate values, the performance of Ek-CA-AODV can closely approach that of the Greedy-AODV algorithm.

6.2 Simulation Study of Combined Channel Assignment and MC-MAC Protocols

A comprehensive simulation study is also performed to evaluate the performance of the proposed MC-MAC protocol combined with distributed channel assignment protocols, such as E2-CA-AODV and CA-OLSR. Three sets of simulations are performed under different network scenarios, namely a one-hop network, a chained network, two grid networks, and two mobile ad hoc networks. In all simulations, the radio range of a node is set to 250 m

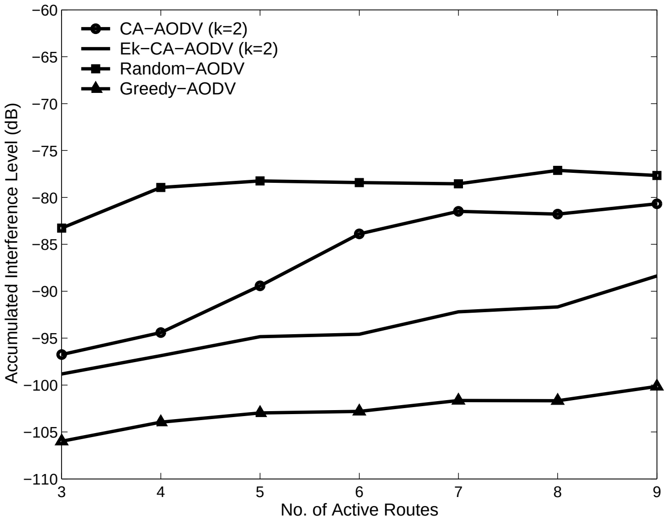


Figure 6.4: Accumulated interference levels at each receiver (CA-AODV and Ek-CA-AODV)

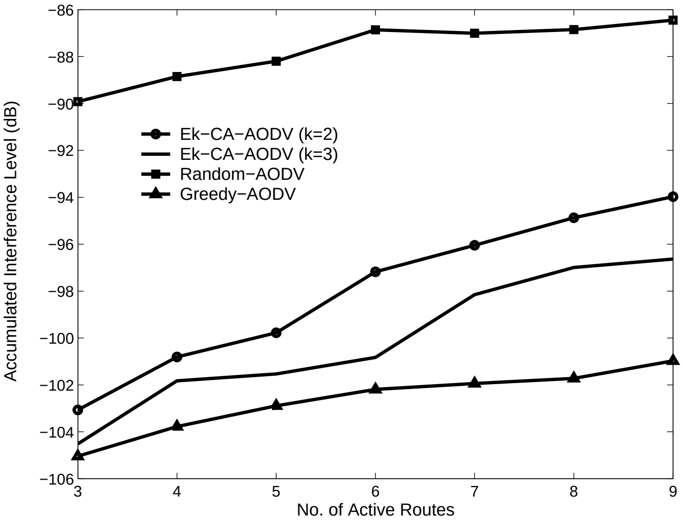


Figure 6.5: Accumulated interference levels at each receiver (Ek-CA-AODV)

and the interference range is set to 550 m, which is approximately twice the radio range. The physical channel bandwidth for all data channels and the control channel are 2 Mbps. For routing, normal AODV and OLSR are used with the IEEE 802.11 MAC, while Ek-CA-AODV, E2-CA-AODV, and CA-OLSR are used with MC-MAC. The warm-up period for all static simulations is set to be 200 seconds, while the warm-up period for all mobile ad hoc simulations is set to be 3600 seconds. The effective simulation period is 500 seconds for static networks and is 4000 seconds for mobile ad hoc networks. The values of the simulation parameters used in the simulations of static wireless networks are summarized in Table 6.3. The static wireless networks include a one-hop network, a chained network, and two grid networks.

Table 6.3: Simulation Parameters Used in the Simulations of Static Wireless Networks

Simulation Parameter	*Value*
Number of nodes	64
Radio range	250 m
Interference range	550 m
Physical channel bandwidth	2 Mbps
Path loss model	Two-ray ground
Channel switch delay	80 μs
Warm-up period	200 seconds
Effective simulation time	500 seconds
Mobility	No

6.2.1 Multi-Channel Capacity of a One-Hop Network

In the first network scenario, 30 wireless nodes are randomly located in a 120 m by 120 m area, such that all nodes are within each other's radio range. UDP flows are introduced between disjoint pairs of nodes. The number of flows varies from 1 to 9. The number of available channels is 13, including a control channel and 12 data channels. The length of a data packet is 1000 bytes. An RTS packet is 40 bytes, CTS and ACK packets are 39 bytes, and the MAC header of a data packet is 47 bytes [47]. Thus, for the IEEE 802.11 MAC, the data throughput, $C_{802.11}$, is at most $1000/(1000+40+39+39+37) \approx 0.8584\ Mbps$. For MC-MAC, data transmissions can occur simultaneously on different data channels. In MC-MAC, RTS and CTS messages carry 1 byte of channel information, so an RTS packet is 41 bytes and a CTS packet is 40 bytes. Data and ACK packets are sent on the data channel, whereas RTS and CTS packets are sent on the control channel. Since all nodes share the same control channel, the control channel may saturate when there are too many data channels in use. In the case of MC-MAC, the control channel will saturate if there are approximately more than $(1000+39+47)/(41+40) \approx 13$ active data channels. This led to our choice of using 12 data channels in the simulation study.

In the one-hop network scenario, Ek-CA-AODV can assign distinct channels to any transmitter when the number of available data channels is sufficiently large. Therefore, as long as the control channel does not saturate, per flow throughput should not decrease much, as confirmed in Figure 6.6, while the system throughput should increase linearly when increasing the number of flows in the network, as confirmed in Figure 6.7(a). When the number of flows is less than the number of available channels, the system throughput of MC-MAC should be $M \cdot C_{802.11}$, where M is the number of flows. Figure 6.7(a) shows that the simulation results are very close to the theoretical capacity limit.

Figure 6.7(b) shows the average packet delay. Note that the average packet delay with MC-MAC is almost constant, while with the IEEE 802.11 MAC there is a sudden jump when the number of disjoint flows increases.

The performance of MC-MAC and that of the IEEE 802.11 MAC are also compared when the OLSR routing protocol is utilized. When the number of data channels is sufficiently large, CA-OLSR can assign distinct channels to any active nodes in the network. Therefore, it can be observed that CA-OLSR with 12 data channels can have the same performance as

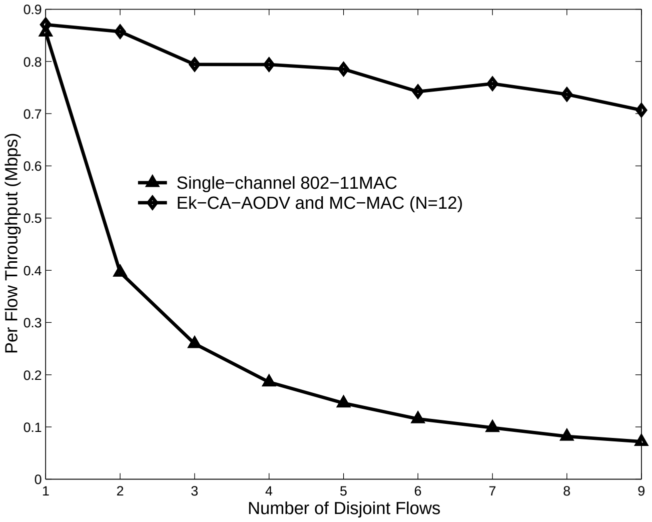
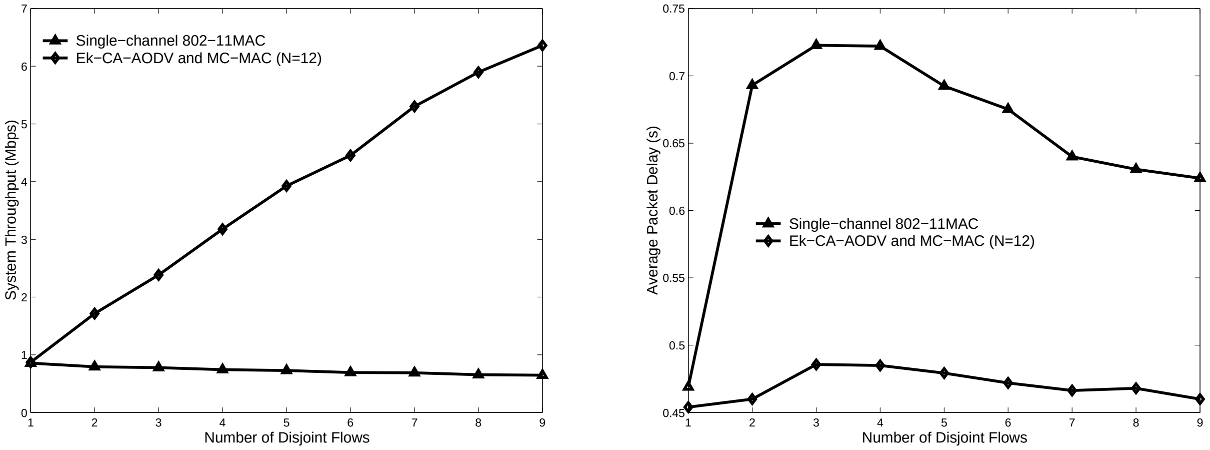


Figure 6.6: Per flow throughput versus the number of disjoint flows for the one-hop network with AODV



(a) System throughput

(b) Average packet delay

Figure 6.7: Performance of Ek-CA-AODV in the one-hop network

the unlimited-channel scheme. However, the throughput drops sharply when there are more active nodes than the number of available channels. For this type of static one-hop network, MC-MAC may suffer significant performance degradation when the number of channels is not sufficient, as illustrated in Figure 6.8 and Figure 6.9(a). The main reason for this performance degradation is that MC-MAC tries to provide fairness to all nodes in the network. If two or more nodes consistently pick the same channel, they will cause significant contention for other nodes because MC-MAC tries to maintain the same transmission probability for all nodes in the network.

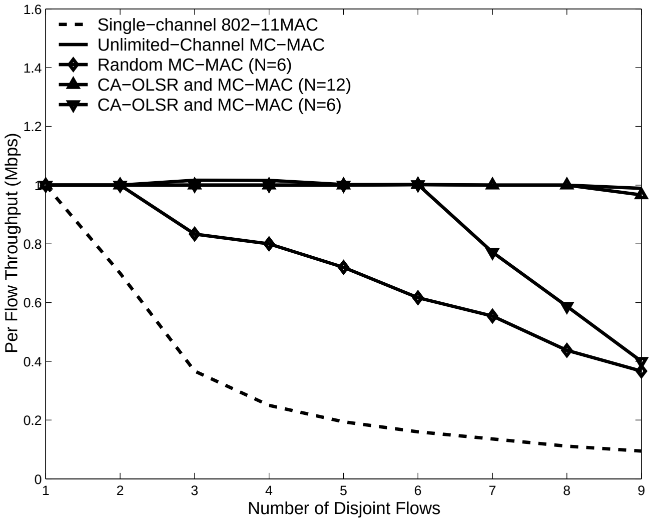
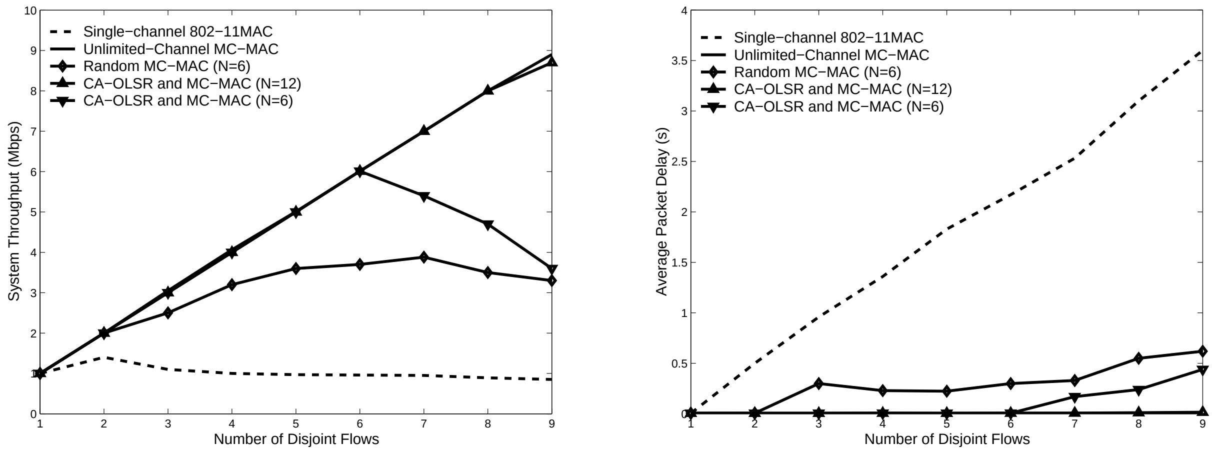


Figure 6.8: Per flow throughput versus the number of disjoint flows for the one-hop network with OLSR

Note that in Figure 6.9(b), the average packet delay of the IEEE 802.11 MAC with OLSR increases almost linearly with an increase in the number of flows, while Figure 6.7(b) shows that, for the IEEE 802.11 MAC with AODV, the delay increases first and then starts to drop. The reason is that AODV is a reactive routing protocol and it is implemented differently from the proactive routing protocol OLSR. In AODV, when the queue at each node is full, a route error message will be sent out, which triggers a new route discovery procedure. Meanwhile the node will empty out its queue to make room for new packets. Therefore, the maximum delay that each packet may suffer is bounded by the queue size. In our simulations, we set the queue size to 50, which means the queue can only accommodate



(a) System throughput

(b) Average packet delay

Figure 6.9: Performance of CA-OLSR in the one-hop network

50 packets. In OLSR, no route error or route broken message will be sent. The queue at each node will not be emptied out either. Instead, packets at the queue will just be delayed. Therefore, the maximum delay that each packet may suffer may grow in an unbounded manner with an increase in the system load.

6.2.2 Multi-Channel Capacity of a Chained Network

In the one-hop network scenario, all traffic is local, i.e., it traverses just a single hop. In this case, MC-MAC offers substantial performance improvement over the single-channel IEEE 802.11 MAC because the capacity of the network increases linearly with an increase in the number of channels. In the second static network scenario, a chained network as shown in Figure 6.12, which consists of multiple nodes is simulated. Nodes are separated by 200 m, which is just under the transmission range. Throughput is examined with both UDP and TCP flows.

Figure 6.10 shows that MC-MAC can achieve at least twice the throughput of the IEEE 802.11 MAC when the chain is at least three hops long. Because each node has only a single half-duplex transceiver, a node cannot transmit and receive at the same time. Therefore,

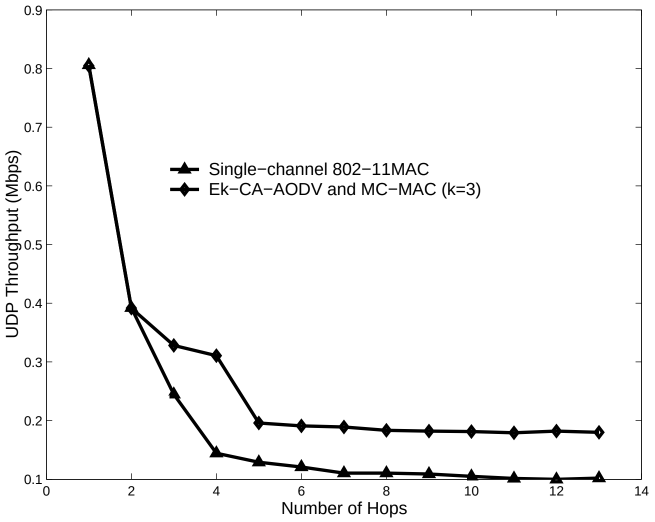


Figure 6.10: UDP throughput versus the number of hops for the chained network utilizing AODV

a two-hop chain network cannot take advantage of multiple channels. When the number of hops is three, data packets can be transmitted on the first hop and on the third hop simultaneously. Similarly, when the number of hops is four or more, data transmissions on hops that are not adjacent to each other can occur simultaneously. This also explains why the throughput of MC-MAC does not degrade much when the number of hops is 2, 3, and 4, whereas the throughput of the IEEE 802.11 MAC drops sharply.

For these results, the neighborhood size k is set to 3, which means that a channel can be re-used every 4 hops. The number of available channels has to be greater than $k+1$. In this case, 6 data channels are assumed to be available.

Ideally, MC-MAC can achieve 50% of the maximum one-hop throughput in a chained network due to the interleaving of data transmission by non-adjacent transmitters. However, simulation results show that the maximum utilization is only around 25% of the maximum one-hop throughput. The discrepancy was also observed by Li, *et al.* [47]. Similar to MC-MAC, the IEEE 802.11 MAC can only achieve about 14% (one-seventh) of the maximum one-hop throughput in a chained network, even though the expected maximum utilization is 25%. Three reasons contribute to this discrepancy. First, both the IEEE 802.11 MAC and

MC-MAC cannot discover the optimal transmission schedule on their own. A slight increase in offered load beyond the optimum can cause the chain throughput to drop sharply [47, 52]. Second, nodes located at the beginning of the chain tend to transmit more packets than subsequent nodes can forward, not only wasting resources, but also preventing transmissions from the subsequent nodes [47]. Finally, control packets that are corrupted by interference may be mistaken as being caused by collisions. Thus, the node's contention window is increased unnecessarily and a certain backoff period is wasted [47]. Because MC-MAC does not have a BEB scheme, the third reason applies to only the IEEE 802.11 MAC.

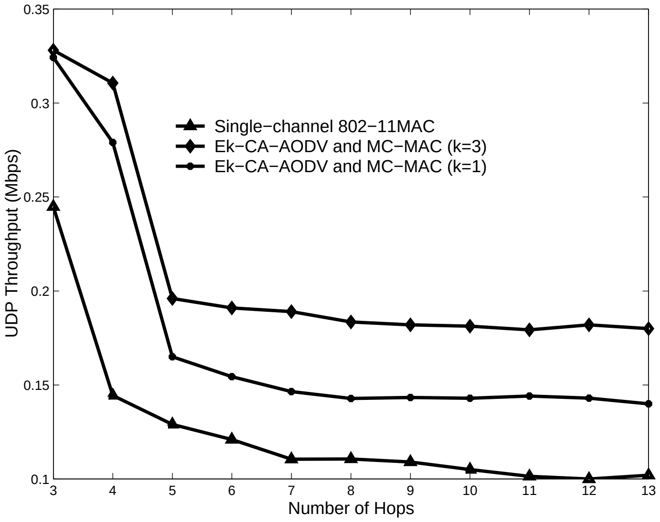


Figure 6.11: UDP throughput for varying neighborhood size, k, for the chained network utilizing AODV

Figure 6.11 illustrates the impact of neighborhood sizes on UDP throughput. When neighborhood size k increases from 1 to 3, the system throughput also improves. In fact, the system throughput will be similar as long as the neighborhood size k is larger than 2 because of the assumption that the interference range is twice the radio range.

Figure 6.12 shows a feasible channel assignment when k=2. The number beneath each node represents the channel assigned to that node. In the case when k is 2, channels are spatially re-used beyond the two-hop range such that no node shares the same channel with another node within its two-hop range. The solid circle represents node D's radio range and the dashed circle shows node D's interference range, which is assumed to be twice as large

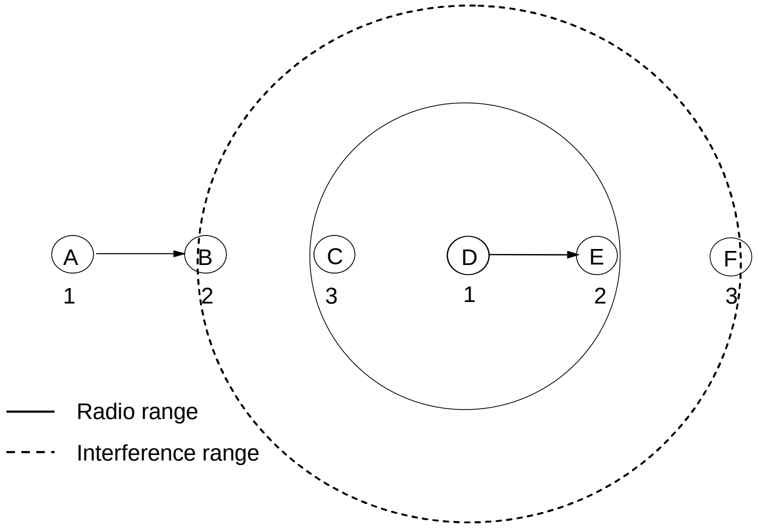


Figure 6.12: A feasible channel assignment for a chained network

as the radio range. When node A transmits to node B, MC-MAC allows node D to transmit to node E. The interference generated by node D's transmission corrupts packet reception at node B. Thus, the system throughput might be reduced. However, because MAC protocols often do not follow the optimal transmission schedule, interference might not be present even when $k = 2$. In fact, for the chained network, when $k = 3$, the performance results are only slightly better than the case when $k = 2$.

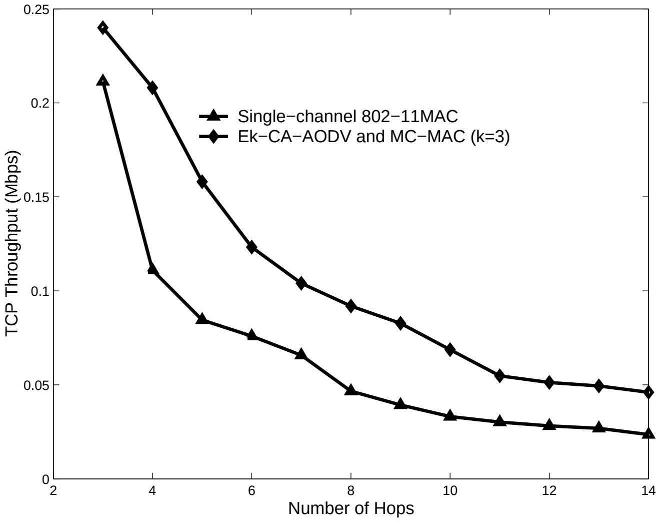
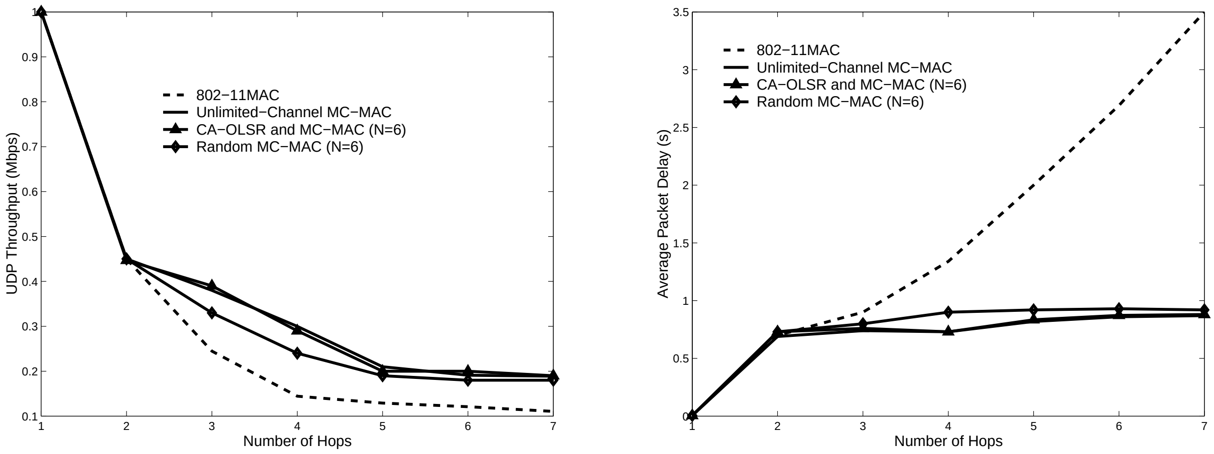


Figure 6.13: TCP throughput versus number of hops for the chained network utilizing AODV

Figure 6.13 illustrates the TCP throughput. It can be noticed that MC-MAC still performs better than the IEEE 802.11 MAC with TCP traffic.

Figure 6.14(a) and Figure 6.14(b) show the throughput and delay, respectively, of MC-MAC and the IEEE 802.11 MAC, when combined with the OLSR routing protocol and carrying UDP traffic. The performance of CA-OLSR combined with MC-MAC is compared against the IEEE 802.11 MAC, a random channel assignment scheme, and a scheme that assumes an unlimited number of channels. In the unlimited data channel scheme, each node has its own unique data channel in the network and a common control channel is shared by all nodes in the network. Nodes that utilize the random algorithm determine their own channel indices based on their MAC addresses. Therefore, no channel assignment is needed in the random scheme. As can be noticed in these two figures, the performance of CA-OLSR combined with MC-MAC is close to that of the random MC-MAC scheme. Additionally, it can be concluded that in a strictly chained network, multi-channel MAC protocols can offer only modest performance advantages over the IEEE 802.11 MAC.



(a) UDP throughput

(b) Average end-to-end packet delay

Figure 6.14: Performance of CA-OLSR in the chained network

6.2.3 Multi-Channel Capacity of Grid Wireless Networks

In the third network scenario, grid wireless networks are considered. Figure 6.15 illustrates that, in a grid wireless network, parallel traffic flows from the left edge to the right edge. There are four chains i interested in grid wireless network erence and inter-flow interference, v rence.

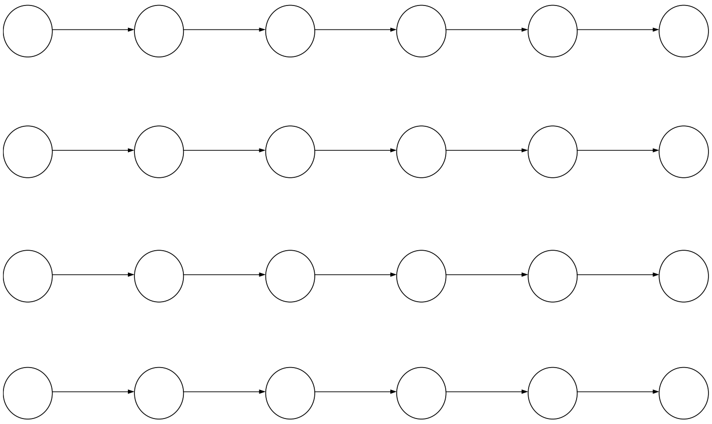
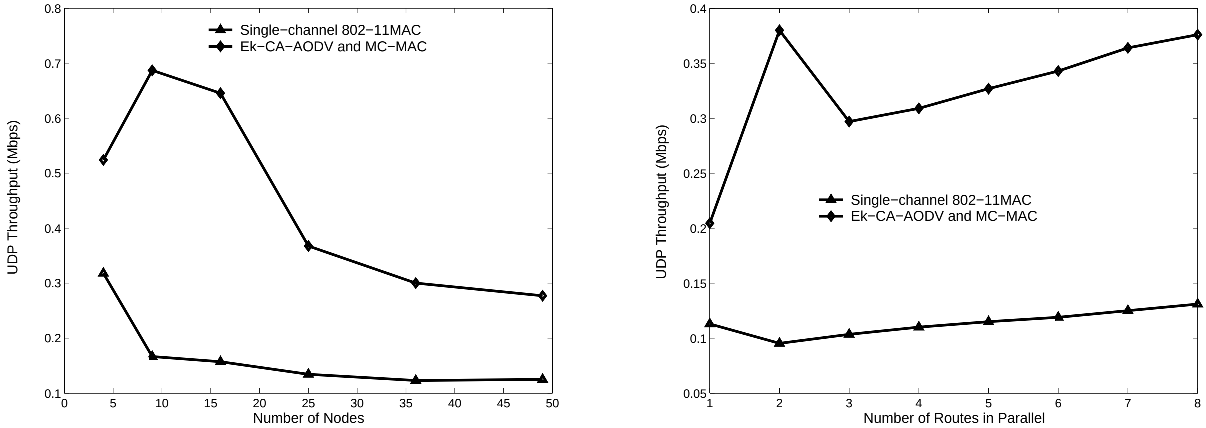


Figure 6.15: An example grid wireless network



(a) A square lattice network

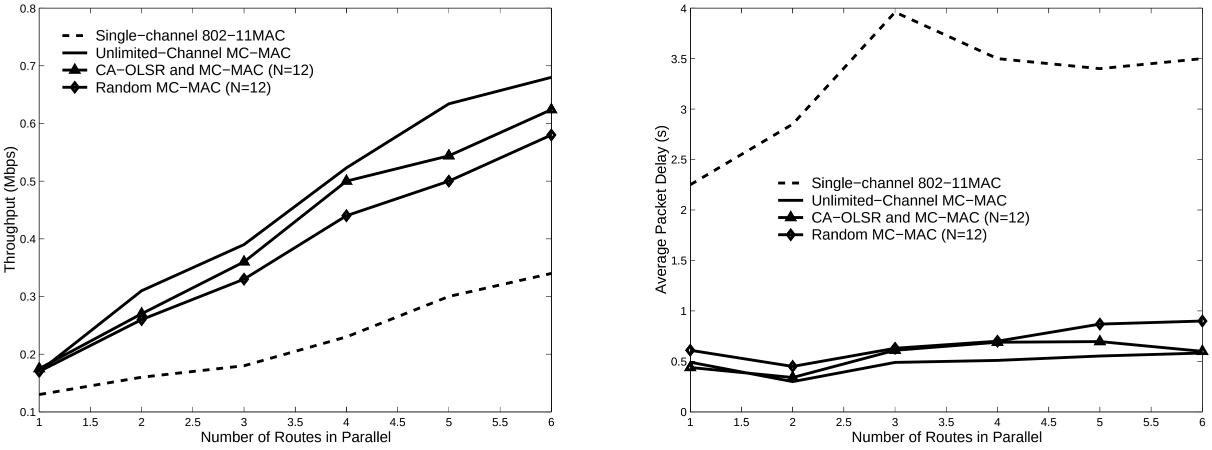
(b) A non-square lattice network

Figure 6.16: Performance of Ek-CA-AODV in grid networks

Figure 6.16(a) compares throughput for the IEEE 802.11 MAC and MC-MAC in a square grid network where the number of chains in parallel is the same as the number of nodes in each chain. For MC-MAC, the number of available data channels is 18 and neighborhood size k is set to 3. For grid networks, the number of channels required is bounded by $(k+1)^2$. Here, 18 data channels (instead of 16 channels) are chosen to introduce more randomness into the simulation. UDP traffic flows in parallel from the left edge of the lattice network to the right edge of the lattice network. The throughput of the IEEE 802.11 MAC decreases as the network size increases, whereas the throughput achieved by MC-MAC increases first and then starts to decrease. These results occur because the IEEE 802.11 MAC has only one channel. In a single-channel ad hoc network, network capacity degrades with the network size [47, 29]. On the contrary, MC-MAC can take advantage, to a point, of the increasing number of channels that can be available as the network size increases. When the network size increases beyond 16 nodes, the throughput starts to drop due to interference introduced by extra nodes and saturation of the control channel.

Figure 6.16(b) shows UDP throughput in a non-square grid network that utilizes AODV. Each chain has 6 hops and the number of chains in parallel increases from 1 to 8. UDP traffic flows from left to right and there is one flow per chain. For MC-MAC, the number of available data channels is 18 and neighborhood size k is set to 3. The throughput of the IEEE 802.11 MAC decreases a little when there are two traffic flows in parallel due to contention introduced by nodes in the second chain. Then, the throughput of the IEEE 802.11 MAC starts to steadily increase as it becomes possible to reuse the channel. MC-MAC achieves the highest throughput when there are two chains due to the additional capacity introduced by multiple channels. MC-MAC's throughput then drops a little due to saturation of the control channel and the inter-flow interference introduced by the third route. When there is only one route, the throughput of MC-MAC is about twice as great as that of the IEEE 802.11 MAC. This is because utilizing multiple channels on a chain can effectively reduce intra-flow interference. When the number of parallel routes increases to two, the throughput of MC-MAC is about four times as large as that of the IEEE 802.11 MAC since MC-MAC combined with Ek-CA-AODV can effectively mitigate both inter-flow interference and intra-flow interference.

Next, I measure the performance of CA-OLSR in the same non-square grid network where each chain has 6 hops and the number of chains in parallel increases from 1 to 8. The performance of CA-OLSR combined with MC-MAC is compared against the IEEE 802.11 MAC, a random channel assignment scheme, and a scheme that assumes unlimited number of channels. Figure 6.17(a) and Figure 6.17(b) show that all three multi-channel schemes have significantly better performance than the single-channel IEEE 802.11 MAC protocol. Further, the performance of CA-OLSR is only slightly better than the random MC-MAC scheme. Therefore, it can be concluded that in grid networks with long parallel routes, intelligent channel assignment may not be necessary.



(a) Aggregate system throughput

(b) Average end-to-end packet delay

Figure 6.17: Performance of CA-OLSR in a non-square grid network

6.2.4 Multi-Channel Capacity of Wireless Ad Hoc Networks with Mobility

In this section, I first compare the performance of E2-CA-AODV with that of an existing channel assignment protocol, specifically the channel assignment scheme (CAS) [23]. CAS is chosen because it is one of just a few distributed channel assignment protocols that can operate in mobile ad hoc networks. I then demonstrate the capacity improvement of MC-

MAC over the original IEEE 802.11 MAC and MC-MAC with randomly assigned channels in a mobile ad hoc network environment.

It is assumed that 64 wireless nodes are placed randomly in a defined area. In all simulations, the radio range of all nodes is set to 250 m and the interference range is set to 550 m, which is approximately twice the radio range. The two-ray ground path loss model [74] is used. The physical channel bandwidth for all data channels and the control channel is 2 Mbps. For multi-channel schemes, it is assumed that there is one common control channel and either 6 or 12 data channels.

For routing, normal AODV or OLSR is used with the IEEE 802.11 MAC, while E2-CA-AODV or CA-OLSR is used with MC-MAC. The first network is an 800 m by 800 m dense network and the second network is an 1600 m by 1600 m sparse network. A dense network is a network where a node has more neighbors and, thus, may experience more collisions and suffer from more interference. In a sparse network, a node has fewer neighbors and suffers from less contention and interference. Mobile nodes move according to the random waypoint model [74]. Both high and low mobility cases are considered. For the high mobility model, the maximum speed is 25 m/s and the minimum speed is 15 m/s. For the low mobility model, the maximum speed is 5 m/s and the minimum speed is 4 m/s. For both mobility models, the maximum pause time is 5 seconds.

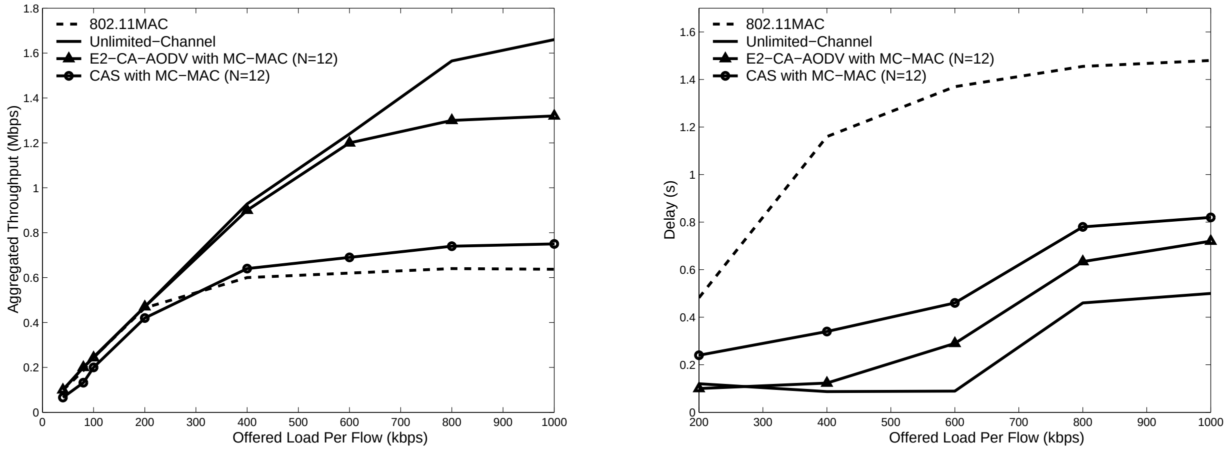
The values of the simulation parameters used in the simulations of two mobile ad hoc networks are summarized in Table 6.4.

E2-CA-AODV is first compared with CAS, the single-channel IEEE 802.11 MAC, and a scheme that utilizes an unlimited number of data channels. In the unlimited data channel scheme, each node has its own unique data channel and a common control channel is shared by all nodes. Proposed by Garcia-Luna-Aceves and Raju [23], CAS is described in Section 4.9 of this dissertation.

Figure 6.18(a) and Figure 6.19(a) show that, because of its high control overhead, CAS performs worse than E2-CA-AODV in both dense and sparse networks. Especially in a dense

Table 6.4: Simulation Parameters Used in the Simulations of Mobile Ad Hoc Networks

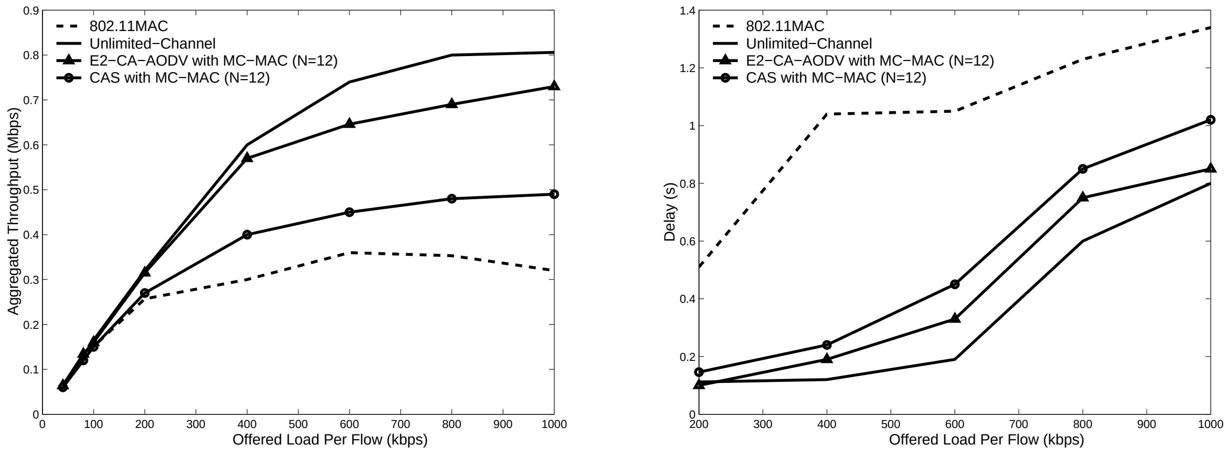
Simulation Parameter	*Value*
Number of nodes	64
Radio range	250 m
Interference range	550 m
Physical channel bandwidth	2 Mbps
Path loss model	Two-ray ground
High mobility model	speed: 18 m/s to 19 m/s, pause time: 5 s
Low mobility model	speed: 4 m/s to 5 m/s, pause time: 5 s
Dense network	800 m by 800 m
Sparse network	1600 m by 1600 m
Available control channel	1
Available data channels	6 or 12
Channel switch delay	80 μs
Warm-up period	3600 seconds
Effective simulation time	4000 seconds



(a) Aggregate system throughput

(b) Average end-to-end packet delay

Figure 6.18: E2-CA-AODV versus CAS in a dense network (low mobility)



(a) Aggregate system throughput

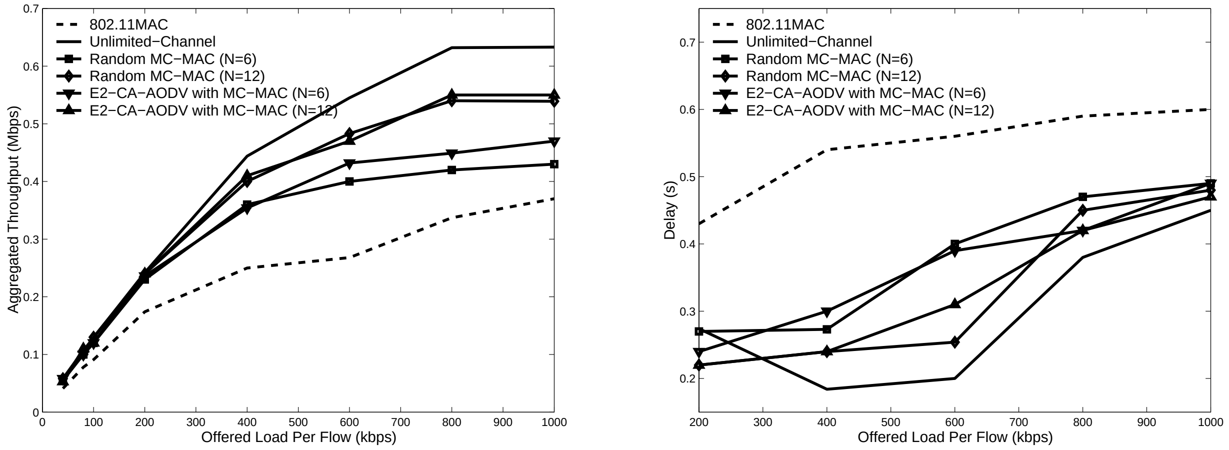
(b) Average end-to-end packet delay

Figure 6.19: E2-CA-AODV versus CAS in a sparse network (low mobility)

network where a node may have many neighbors, the control overhead of CAS is so high that when the data rates are lower than 400 kbps, CAS with 12 data channels performs even worse than the IEEE 802.11 MAC. Figure 6.18(b) and Figure 6.19(b) show that the delay performance of CAS is better than that of the IEEE 802.11 MAC protocol, but still not as good as that of E2-CA-AODV. By utilizing multiple channels, CAS can effectively reduce collisions and contention in the network. Thus, the end-to-end delay suffered by a data packet is reduced.

The E2-CA-AODV protocol is compared with a scheme that utilizes an unlimited number of data channels and a random algorithm. Nodes that utilize the random algorithm determine their own channel indices based on their MAC addresses. Therefore, no channel assignment is needed in the random scheme.

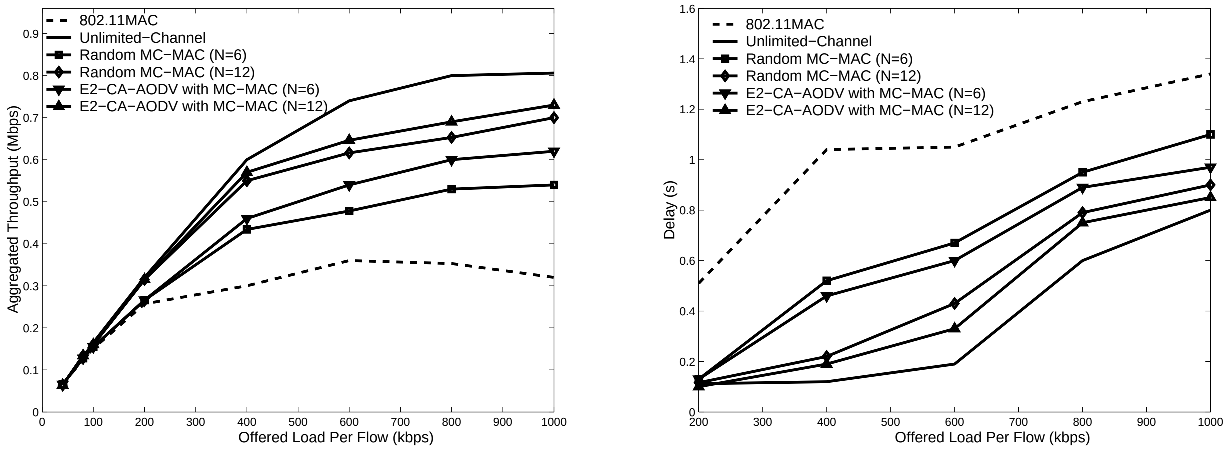
Figure 6.20(a), Figure 6.21(a), Figure 6.22(a), and Figure 6.23(a) show that E2-CA-AODV performs better than the random scheme with respect to throughput for a range of values for network density and mobility. Figure 6.23(a) shows that the performance gap between E2-CA-AODV and the random scheme increases when the number of available channels decreases. Figure 6.23(a) shows that E2-CA-AODV combined with MC-MAC can have up to three times higher throughput than the IEEE 802.11 MAC. Figures 6.20(b), 6.21(b), 6.22(b), and 6.23(b) show that the end-to-end delay increases for all schemes when the data rate increases. However, multi-channel schemes utilizing 6 or 12 data channels have lower delay than the original IEEE 802.11 MAC scheme.



(a) Aggregate system throughput

(b) Average end-to-end packet delay

Figure 6.20: Performance of E2-CA-AODV in a sparse network (high mobility)

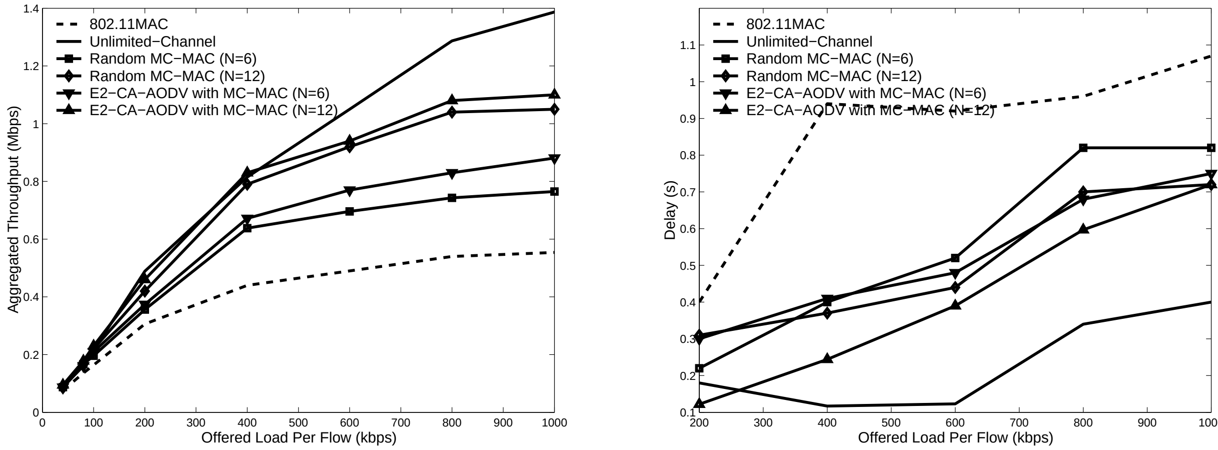


(a) Aggregate system throughput

(b) Average end-to-end packet delay

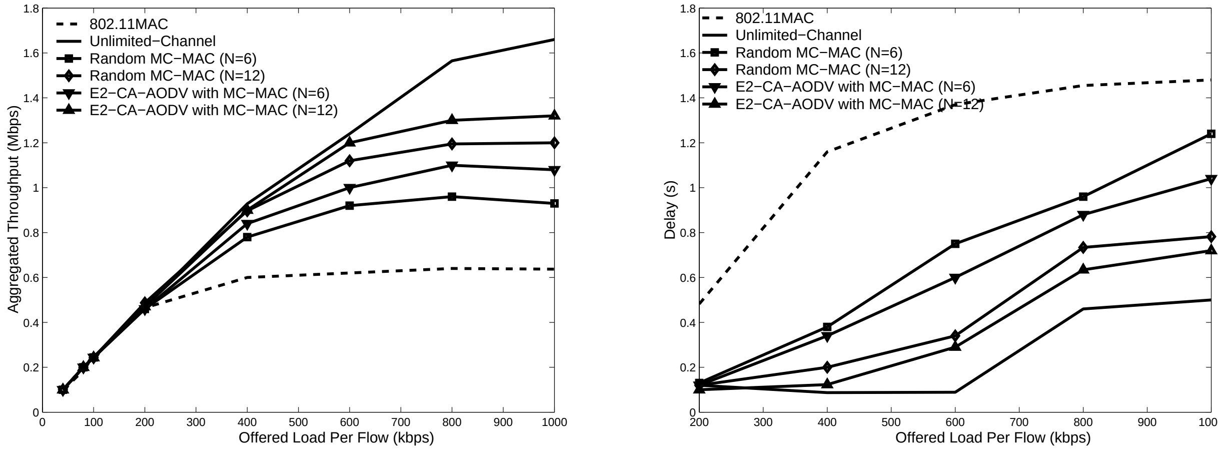
Figure 6.21: Performance of E2-CA-AODV in a sparse network (low mobility)

First, CA-OLSR is compared against CAS combined with MC-MAC. Figure 6.24(a) and Figure 6.25(a) show that because of its high control overhead, CAS has lower system throughput than CA-OLSR in both dense and sparse networks.



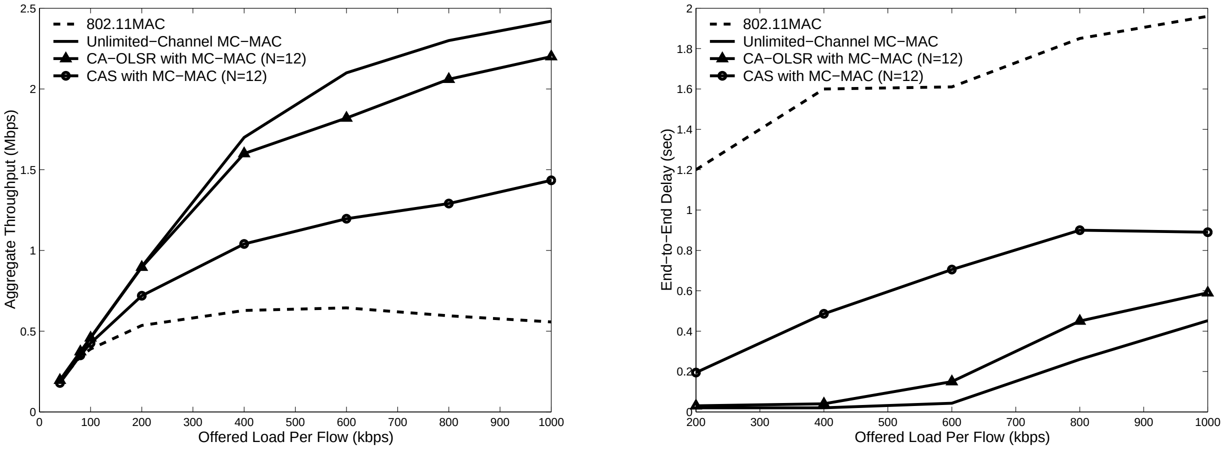
(a) Aggregate system throughput (b) Average end-to-end packet delay

Figure 6.22: Performance of E2-CA-AODV in a dense network (high mobility)



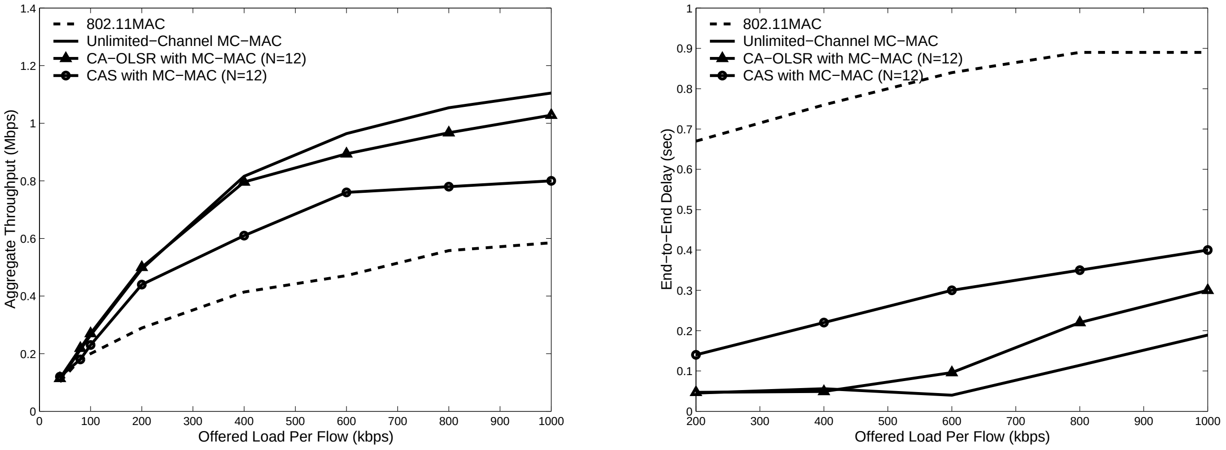
(a) Aggregate system throughput (b) Average end-to-end packet delay

Figure 6.23: Performance of E2-CA-AODV in a dense network (low mobility)



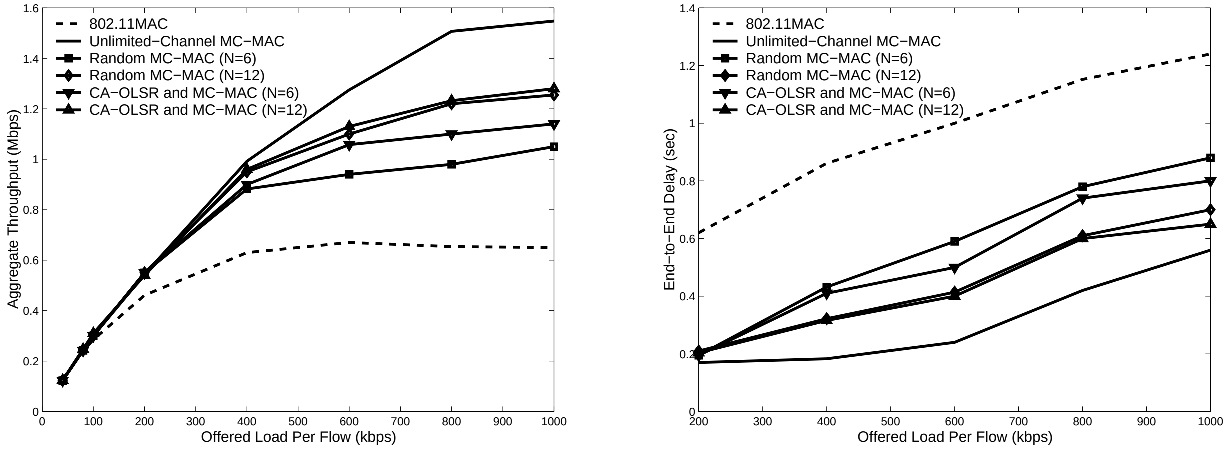
(a) Aggregate system throughput (b) Average end-to-end packet delay

Figure 6.24: CA-OLSR versus CAS in a dense network (low mobility)



(a) Aggregate system throughput (b) Average end-to-end packet delay

Figure 6.25: CA-OLSR versus CAS in a sparse network (low mobility)



(a) Aggregate system throughput

(b) Average end-to-end packet delay

Figure 6.26: Performance of CA-OLSR in a dense network (high mobility)

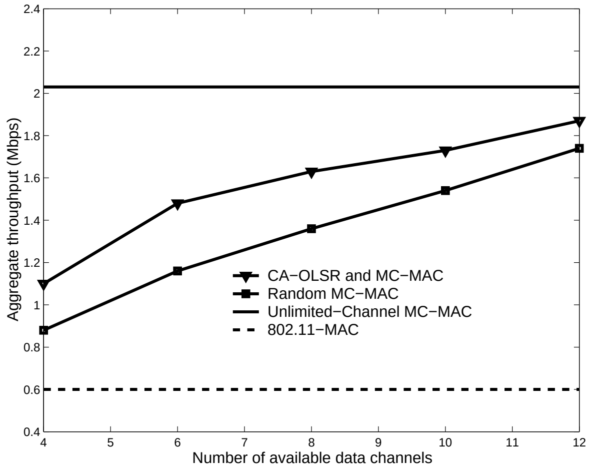
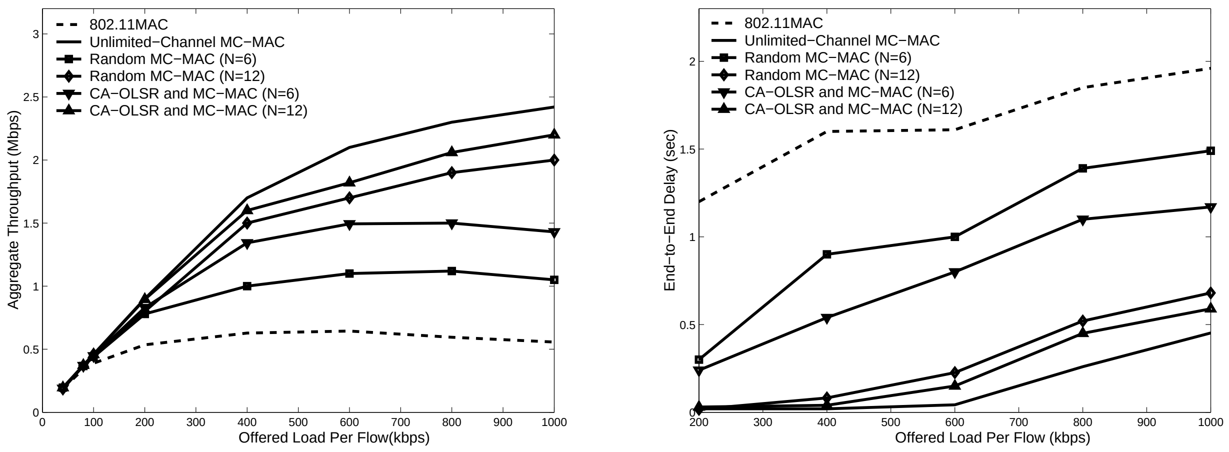


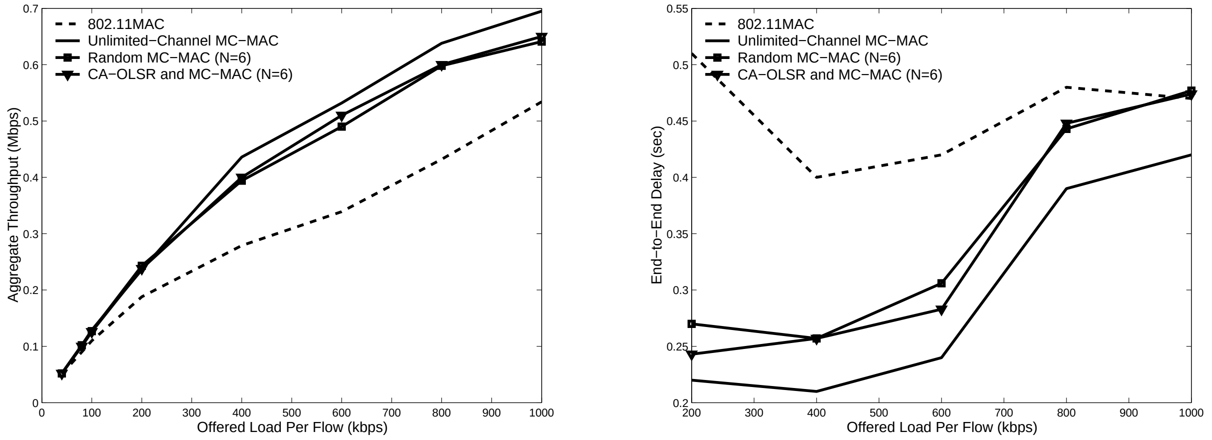
Figure 6.27: Impact of the number of channels on the aggregate throughput in a dense network (low mobility)



(a) Aggregate system throughput

(b) Average end-to-end packet delay

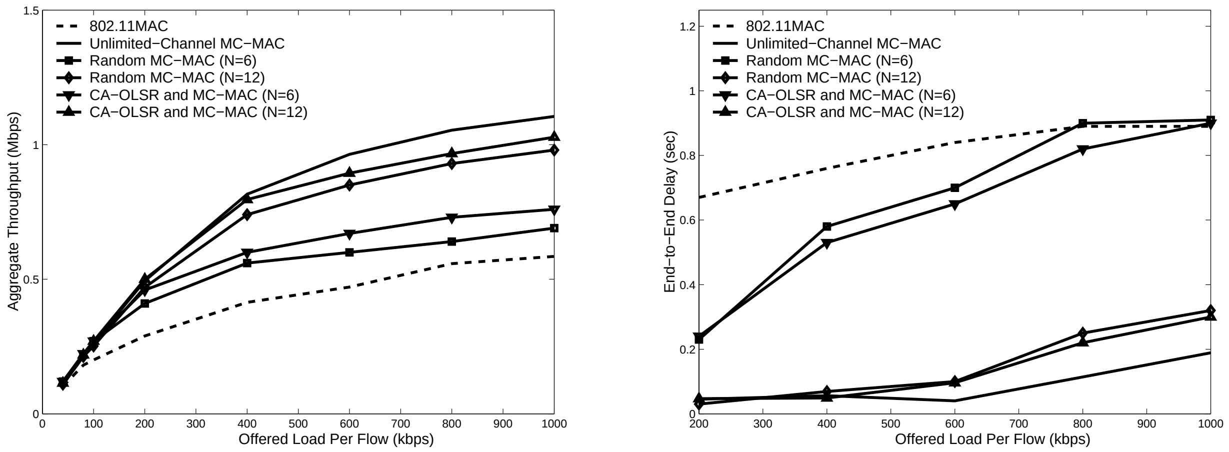
Figure 6.28: Performance of CA-OLSR in a dense network (low mobility)



(a) Aggregate system throughput

(b) Average end-to-end packet delay

Figure 6.29: Performance of CA-OLSR in a sparse network (high mobility)



(a) Aggregate system throughput

(b) Average end-to-end packet delay

Figure 6.30: Performance of CA-OLSR in a sparse network (low mobility)

Then, CA-OLSR is compared with the random channel assignment scheme, the single-channel IEEE 802.11 MAC, and the unlimited channel scheme. Figure 6.28(a) shows that CA-OLSR performs better than the random-OLSR scheme in a dense network for all cases studied, given an equal number of channels. The performance gap between the two increases as the number of available channels decreases. Note that CA-OLSR combined with MC-MAC can have up to four times higher throughput than the IEEE 802.11 MAC. Figure 6.28(b) shows that the end-to-end delay increases for all schemes when the data rate increases. However, multi-channel schemes that utilize 12 data channels have much lower delay than the original IEEE 802.11 MAC scheme.

Figure 6.30(a), Figure 6.30(b), Figure 6.29(a), and Figure 6.29(b) show the throughput and delay for all schemes in a sparse network. The performance gaps between different schemes are not as large as those in the dense network, although CA-OLSR still has an advantage over the random-OLSR scheme. In a sparse network, the interference and the likelihood of collisions caused by neighboring nodes is lower than in a dense network because, in a sparse network, each node has fewer neighbors and the distance between neighbors is greater. Therefore, the performance gain of multi-channel MAC schemes is not as significant in a sparse network. Figure 6.27 shows the performance improvement with an increase in the number of available data channels in a dense network. The performance gap between CA-OLSR and random-OLSR decreases when the number of available channels increases. When mobility is high, the routes stay connected only for a short period of time. Thus, when nodes in a sparse network have high mobility, utilizing multiple channels does not offer significant performance advantage.

Based on the simulation results presented in this section, the following nine observations are made.

1) Communication overhead has a profound impact on the performance of distributed channel assignment protocols. E2-CA-AODV, CA-OLSR, and CAS all seek to assign distinct channels to nodes in a two-hop neighborhood. However, because E2-CA-AODV and CA-OLSR have lower communication overhead, they have much better performance than CAS, especially in a dense network.

2) As expected, if the control overhead is not prohibitively high, utilizing multiple channels gives better performance than the IEEE 802.11 MAC scheme. Utilizing multiple channels increases the capacity available in an ad hoc network.

3) The network density has a much greater impact on the performance than the mobility. The performance gap between multi-channel schemes increases when the network density increases. This is because more interference is generated from the greater number of neighboring nodes in a dense network. Therefore, intelligent channel assignment has a bigger advantage over random channel assignment in a dense network than in a sparse network. Mobility does not significantly change the interference level or contention in a network.

4) When the offered load is very low, the multi-channel schemes do not offer a performance advantage over the IEEE 802.11 MAC protocol. This implies that utilizing multiple channels may not be necessary for low data rate applications.

5) In all simulated scenarios, the performance gap between the scheme with an unlimited number of channels and E2-CA-AODV and CA-OLSR with 12 data channels is not significant. Thus, it is concluded that, due to the negligible interference generated by nodes that

are far away, a large number of data channels is not necessary to gain benefits.

6) The performance gap between E2-CA-AODV and CA-OLSR and the random scheme decreases when the number of available channels increases. E2-CA-AODV and CA-OLSR assign channels more intelligently than the random scheme. I showed in Section 4.8 that Ek-CA-AODV and CA-OLSR can assign distinct channels to any active node in a two-hop neighborhood when the number of available channels is sufficiently large. Therefore, CA-OLSR can effectively avoid primary collisions and secondary collisions, while E2-CA-AODV can avoid collisions and mitigate intra-flow interference. The random scheme merely tries to utilize all available channels without intelligent channel assignment. However, when a large number of available channels are available, collisions may be infrequent even if channels are assigned randomly.

7) The performance advantage of E2-CA-AODV and CA-OLSR combined with MC-MAC over the IEEE 802.11 MAC scheme is not in proportion to the number of channels utilized. For instance, for MC-MAC with 12 available data channels, the throughput gain can be up to a factor of four compared to the throughput achieved by the IEEE 802.11 MAC. The question is whether there is a way to get close to 12 times the performance gain when there are 12 available data channels. The answer is yes, if there are a sufficient number of transceivers at each node, traffic flows can utilize the channels, and the control channel does not saturate.

Most current mobile devices are equipped with only one half-duplex transceiver. I want to study the performance gain that an off-the-shelf mobile device can obtain using multiple available channels. Therefore, I assume the same hardware requirement for both MC-MAC and the IEEE 802.11 MAC. For MC-MAC, each node is assumed to be equipped with only one half-duplex transceiver, which is time-shared between transmitting and receiving, and between the common control channel and the data channel. With an increase in the number of data channels, the time period allocated to the common control channel also increases because the number of collisions on the common control channel increases as more nodes try to transmit in parallel. Therefore, the time that a transceiver can spend transmitting data packets is reduced, which degrades the performance advantage of CA-OLSR. In addition, when there is only one half-duplex transceiver available at each node, the channel switch delay is non-negligible, which further degrades the performance. In fact, Kyasanur and Vaidya [45] show that when the number of transceivers is less than the number of available channels, network capacity degrades in many cases.

8) Compared with the single-channel MAC scheme, the performance improvement of MC-MAC in terms of end-to-end delay is more significant than the performance improvement in terms of throughput. There are two reasons for this. First, due to the reduction in the number of collisions and backoffs, the packet delay at each node is reduced. Secondly, a route from a source to a destination usually consists of multiple nodes. The end-to-end packet delay is accumulated over all hops along the same route. Because the same data packet experiences delay reduction at each hop, the reduction in end-to-end delay can be significant.

9) E2-CA-AODV can offer up to a factor of three performance improvement over the IEEE 802.11 MAC protocol, while CA-OLSR can offer up to a factor of four performance improvement over the IEEE 802.11 MAC protocol. There might be two reasons as to why CA-OLSR performs better than E2-CA-AODV. First, OLSR, as a proactive routing protocol, may perform better than a reactive routing protocol, AODV, in a mobile ad hoc environment due to OLSR's low control overhead in mobile scenarios. Secondly, the control messages, such as `HELLO` messages, are exchanged more frequently in OLSR than in AODV. Thus, more up-to-date topology and neighbor information is available in OLSR, which helps to assign channels correctly.

6.3 Summary

This chapter presents the performance evaluation of combined channel assignment and MC-MAC protocols. Preliminary simulation results show that the performance of the proposed distributed channel assignment protocols can sometimes approach that of a greedy algorithm, which is a near-optimal algorithm. A comprehensive simulation study shows that, even with only a single half-duplex transceiver, the combined channel assignment and MC-MAC protocols can offer up to a factor of four improvement in throughput over the IEEE 802.11 MAC protocol.

Chapter 7

An Architecture for Cross-layer Design

In this chapter, the cross-layer design principle is applied to more general network design problems. Though cross-layer design is not a new concept, a unified framework for accommodating a variety of important cross-layer design techniques does not exist. Thus, I propose a cross-layer design architecture that facilitates interactions and information exchange among different layers. Additionally, a survey of many existing cross-layer design techniques is presented to show how these techniques can be mapped to the proposed cross-layer design architecture.

7.1 Description of a Cross-layer Design Architecture

Traditionally, network protocol designs follow a strict layered approach, where each layer can exchange information with only the layer directly above it or the layer directly beneath it. However, a strict layered approach can often lead to performance degradation, especially in time-varying wireless networks. A cross-layer design architecture is proposed to not only support adaptation or optimization at a single layer, but also to allow multiple layers to

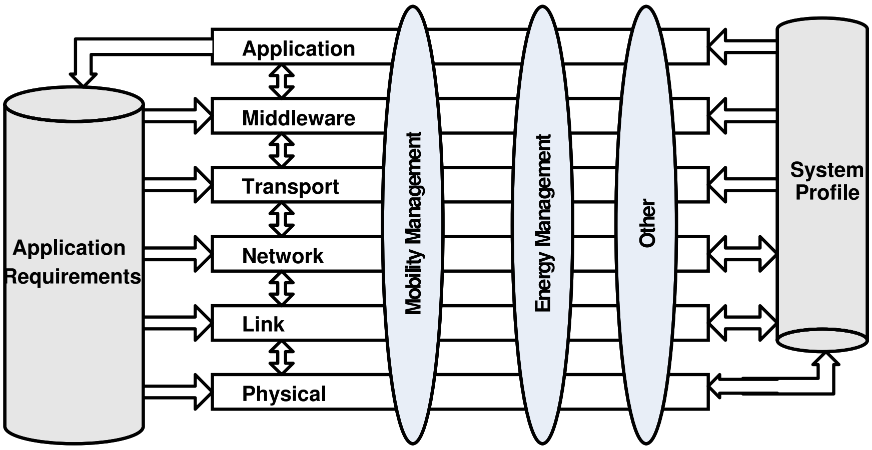


Figure 7.1: An architecture for cross-layer design

jointly optimize one or more performance metrics.

For instance, most research on power-aware routing has been focused on a cross-layer approach, where the physical layer provides transmission power information or remaining battery power information to the network layer. The network layer then makes a decision to pick either minimum energy routes or routes that can extend battery life. However, this approach has largely ignored power conservation issues at the MAC layer, where collisions and re-transmissions can also waste battery power. Further, higher layers, such as the application layer, may act adaptively to save power as well as to maintain acceptable system performance. In this case, cross-layer optimization techniques that jointly optimize as many layers as necessary may provide more significant performance benefits. Currently, there is no cross-layer network architecture to help researchers identify layers that may affect the overall system performance in terms of a particular performance metric or to assist in formulating a structure for a design.

Inspired by Conti, *et al.* [15], and Goldsmith and Wicker [24], I propose a cross-layer architecture, Figure 7.1, that involves six layers in the overall protocol stack, i.e., application layer, transport layer, network layer, link-layer, middleware layer, and physical layer. The middleware layer is included in the cross-layer network architecture because it plays an important role in some cross-layer design schemes. Note that the middleware layer may not implement a protocol, but rather it provides functions on each node using protocols on other layers.

Two global data structures are maintained in the proposed architecture to facilitate information exchange among all six layers. One is the system profile data structure and the other is the application requirements data structure. Application requirements are normally throughput or QoS requirements. Layers in the cross-layer design architecture react both locally and globally to the current system profile and application requirements. The system profile includes network and channel information measured by the three lowest layers, namely the physical layer, the data link layer, and the network layer. For instance, link status, node position, remaining battery power, route congestion status, and frame error rate (FER) can all be part of the system profile.

The application requirements derived from the application layer are used in coordinating the behavior of the lower layers, which map the application requirements into their own set of parameters. The application layer adjusts its QoS requirements based on the current system profile. This type of behavior is sometimes called *soft QoS* adaptation [44]. Because it is challenging to provide static QoS guarantees in a wireless network, "soft" QoS requirements are adopted in the proposed cross-layer design architecture. Each layer may interpret QoS requirements differently. For instance, at the application layer, QoS can be measured by end-to-end packet delay, jitter, mean squared error (MSE) of a packet, and peak-signal-to-noise ratio (PSNR) if a packet. The network layer may map the requirements derived from the application layer into packet priorities for scheduling purposes. At the data link layer, QoS is often expressed in terms of the probability of buffer overflow and the probability of packet delay violation. The physical layer may choose different modulation schemes to satisfy different data rate requirements.

While each layer may adapt locally to perform its assigned functions, some management entities require global coordination across two or more layers. For instance, mobility management and energy management may require global adaptation across all layers, while other management entities, such as routing management, may need cross-layer optimization across two or three layers. The key to cross-layer design is a holistic view of the network and interactions among layers in the protocol stack. Another essential factor to adaptation is the system's ability to estimate the current and even predict the future network and channel conditions.

It is worth pointing out that implementing cross-layer design techniques in systems and prototypes is not an easy task, especially for cross-layer design techniques that involve more than two layers or need coordination between software and hardware. For instance, most of the data link layer functions and physical layer functions are implemented in hardware, whereas functions of higher layers, such as network layer, transport layer, middleware, and application layer, are implemented in software. How to effectively exchange information between hardware and software is an important practical design issue.

7.2 Current Cross-Layer Design Research

Depending on the techniques used, cross-layer design can be categorized into two major classes: joint-layer design using optimization techniques and adaptive cross-layer design.

Some prior work that is discussed in this section jointly designs two or three layers using nonlinear optimization techniques, whereas other prior work applies adaptive techniques on one layer or across multiple layers based on information provided by other layers. Within the adaptive cross-layer design category, some techniques adapt to the system profile, while others seek to provide end-to-end QoS in an unreliable wireless environment.

7.2.1 Joint Layer Design Using Optimization Techniques

Many researchers have proposed to jointly design two or more layers, using nonlinear optimization techniques. While the techniques used in this body of research are highly diverse, the majority of the techniques focus on jointly designing power control at the PHY layer and

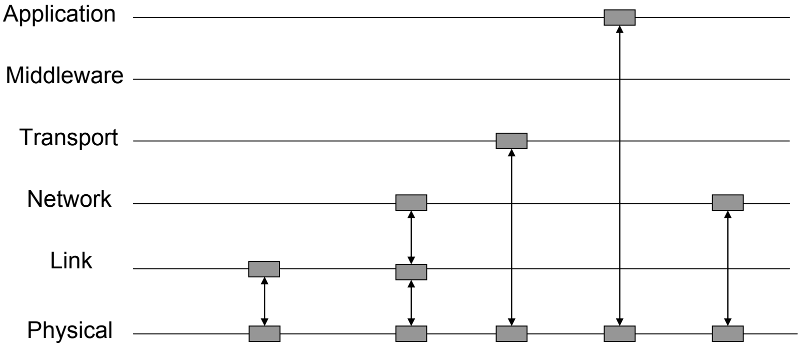


Figure 7.2: Joint-layer design using optimization scheme for power control

one or more higher layers, as shown in Figure 7.2. Power control has been a popular topic for years, mainly because it can be exploited to reduce transmit power, thus extending the battery life of wireless devices, suppressing multi-user interference, and increasing network throughput [43].

For instance, Elbatt and Ephremides [18] introduce a cross-layer design framework that combines power control at the physical layer with time-division multiple access (TDMA) scheduling at the MAC layer, with the objective of scheduling the maximum number of links in the same time slot. In other words, the goal is to maximize the per hop throughput of the network. Due to the complexity of the problem, the authors break the algorithm into two phases, namely, a scheduling phase and a power control phase. The scheduling phase seeks to schedule as many simultaneous transmissions as possible, while still achieving the desired signal-to-interference-noise-ratio (SINR). In the power control phase, two distributed power control algorithms are formed within the context of TDMA and hybrid TDMA and CDMA wireless ad hoc networks. The authors show that distributed power control algorithms used for uplink control in cellular networks can be applied under the proposed system model and, thus, the iterative algorithm developed for cellular networks can be employed in wireless ad hoc networks as well. However, the system model proposed in [18] does not consider a multi-hop wireless environment.

Cruz and Santhanam [16] propose a different approach to jointly design power control and scheduling. The goal is to minimize the long term average power consumption in the network, while still providing long term end-to-end rate guarantees to a set of sessions. One of the main assumptions made is that when the system operates at significantly low SINR levels, the data rate on each link can be approximated as linearly dependent on SINR. Therefore, the transmit power can be used directly as a throughput guarantee constraint, rather than as a quality of service guarantee for bit error rate (BER). It is also assumed that the transmitters are time-synchronized and channel conditions remain constant over several time slots. After a dual problem is solved, the resulting optimal power control policy reveals that, in a particular time slot, a link has to either transmit at the maximum allowed power or not transmit at all. In addition, the authors extend the cross-layer approach to the routing layer and find that minimum-energy path routing combined with TDMA scheduling is near-optimal in low noise regimes and when the required average data rates are small. Thus, an optimal joint routing, scheduling, and power control policy is obtained. However, it is worth pointing out that in a purely distributed ad hoc environment where no central control node is available, it is very challenging to synchronize all of the nodes. Thus, a TDMA MAC protocol may not be practical.

As illustrated by our architecture for cross-layer design, a good design for power management should consider as many layers as possible. For instance, in order to minimize total average power consumption, Cruz and Santhanam [16] take three layers into consideration, namely, the PHY layer, MAC layer, and routing layer. Kozat, Koutsopoulos, and Tassiulas [43], in contrast, propose a different approach for energy-efficient communications. They address the problem of joint power control and scheduling with the objective of minimizing the total transmit power, subject to the end-to-end QoS guarantees for sessions in terms of their allocated bandwidth and BER. Because the resulting feasibility problem is $\mathcal{NP}$-complete, a joint feasible allocation becomes intractable. Thus, the authors devise two heuristic algorithms that use greedy approaches. However, these two algorithms are centralized and require global network knowledge. Moreover, the performance gap between the heuristic algorithms and the performance limits of wireless multihop networks remains unknown.

To increase the end-to-end throughput and network energy efficiency, Chiang [13] proposes to jointly optimize power control and congestion control. The problem is formulated as a nonlinear constrained optimization problem, with the objective of maximizing network utility subject to an elastic link capacity constraint. Network utility is represented through a continuously differentiable, increasing, and strictly concave function of source data rate x, while the link capacity constraint relates to both the source rate x and the transmit power P. To jointly optimize the source rate and the transmit power, a distributed power control algorithm that is closely coupled with TCP Vegas is derived. It is also proved that the nonlinearly coupled system converges to the global optimum of the joint optimization problem.

Another optimization problem is formulated by Lu, *et al.* [49], based on models of video distortion [68] and power consumption, where the goal is to minimize the total power consumed for video transmissions at both the application layer and at the physical layer. The total power consumption incorporates the source compressor, channel coder, and the power consumed by the transmitter. Because the problem space is small, an exhaustive search procedure is performed to jointly reduce the power dissipated by the source compressor and the power used to transmit the compressed bits through the channel, subject to a constraint on the end-to-end distortion.

To improve system performance, a joint routing and power allocation problem in CDMA systems is studied in [37]. For TDMA and FDMA systems, the link capacity may be assumed to be a concave and increasing function of resources allocated to the link by Johansson, Xiao, and Boyd [77]. However, the same assumption cannot be held true in CDMA systems, due to the nonorthogonality of CDMA channels. Because the capacity constraints are no longer jointly convex with respect to data rate and power allocation, the non-convex problem becomes hard to solve. Thus, the authors propose to use coordinate projection to convert the problem into an equivalent convex optimization problem whose solution is guaranteed to be feasible for the original non-convex formulation. Lastly, a heuristic link removal procedure is proposed to further improve the system performance.

Table 7.1: An Overview of Joint-Layer Design Techniques

Ref.	Description	Goal
[18]	Power control and scheduling	Maximize per hop throughput of the network
[16]	Power control, scheduling, and routing	Minimize long term average power consumption in the network
[43]	Power control and scheduling	Minimize total transmit power
[13]	Power control and congestion control	Maximize network utility (i.e. a function of source data rate)
[49]	Power control and video distortion control	Minimize total power consumed for video transmission
[37]	Routing and power allocation in CDMA networks	Improve system performance via minimizing a cost function

The joint-layer design techniques introduced in this section are summarized in Table 7.1.

7.2.2 Cross-layer Design Based on Adaptive Techniques

Even though joint-layer design techniques can achieve globally optimum solutions, they suffer from high complexity and may have convergence problems. In addition, unrealistic or over-simplified assumptions are often used in the problem formulation. Thus, joint-layer design techniques are often not practical. Adaptive techniques, however, may achieve relatively good performance, while at the same time, are relatively easy to implement.

Adaptive cross layer designs may take two different approaches: system-profile-adaptive or QoS-driven. As stated in Section 7.1, system profile information includes node position, channel conditions, link status, network congestion level, etc. Because of the randomness in wireless channels, most of the cross-layer design techniques used in wireless networks are system-profile adaptive. However, even in traditional wired networks, QoS-driven techniques

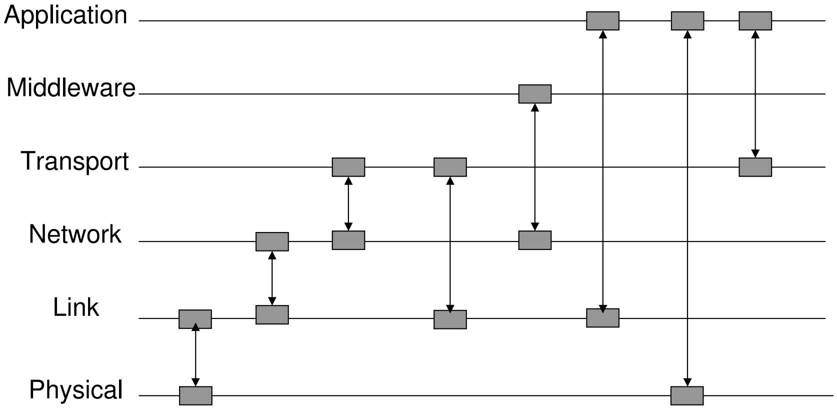


Figure 7.3: Adaptive cross-layer design schemes

are inherently cross-layer designs because layers underneath the application layer have to provide network support to meet QoS requirements for different traffic classes. In an unreliable wireless environment, both system-profile and QoS requirements need to be taken into consideration to provide end-to-end QoS. Figure 7.3 gives a representation of different adaptive cross-layer design schemes.

The main objective of system-profile-adaptive designs is to maximize the network throughput, minimize packet delivery latency, or minimize power consumption. In contrast, QoS-driven cross-layer designs seek to improve robustness and efficiency of voice or video transmissions over time-varying and non-stationary wireless channels.

Cross-layer Designs Adaptive to System Profile

Since the PHY layer and the MAC layer are often closely coupled together, even in traditional data networks, much of the prior work has focused on MAC/PHY cross-layer design [50, 31, 58, 76, 65].

Maharshi, Tong and Swami [50] explore the possibility of utilizing MAC parameters, such as spreading codes, at the physical layer and, in the reverse direction, the measurements at the PHY layer in the MAC acknowledgement. Holland, Vaidya, and Bahl [31] propose a receiver-based rate adaptation scheme is proposed to improve WLAN performance in terms of throughput and packet delivery ratio. MAC layer RTS and CTS control messages are used to exchange physical layer information. Based on the measured signal strength of a RTS message, the receiver selects a modulation scheme with the highest allowable data rate under the current estimated channel condition and, then, sends the selected data rate to sender in the CTS message. To improve the goodput of WLAN, a technique called "link adaptation" is proposed by Qiao and Choi [58]. Based on measured channel conditions, their technique performs both dynamic fragmentation at the link layer and rate selection at the physical layer. The adjustable parameters include modulation type, code rate, and MAC fragment size. Toumpis and Goldsmith [71] study the performance of wireless ad hoc networks in a multi-hop traffic environment, with the emphasize on the MAC layer. Two contention based MAC protocols are proposed, including a progressive back off algorithm (PBOA) that performs medium access jointly with power control at PHY layer and a progressive ramp up algorithm (PRUA).

Multi-channel MAC schemes allow the sender and receiver to exchange physical layer information using MAC control messages [76, 65]. These messages carry information about available physical channels and are exchanged between a sender and a receiver. If the sender and receiver agree on a specific data channel, they will both switch to this channel for data transfers that will follow. Wu, *et al.* [76] propose a Dynamic Channel Assignment (DCA) protocol, where there are multiple data channels and one shared control channel in the system. Each host is assumed to have two transceivers so that it can listen on the control channel and the data channel simultaneously. A sender includes a list of preferred channels in RTS messages. Upon receiving a RTS, the receiver decides on a channel and sends back a CTS with the chosen channel information. Then, a data packet and an ACK message are exchanged using the agreed upon data channel. So and Vaidya [65] propose another multichannel MAC (MMAC) protocol that only requires one receiver for each node. They assume that nodes are time-synchronized so that all nodes can start their beacon intervals at the same time. In a predefined window at the beginning of the beacon interval, nodes send and listen on a common channel to negotiate channels. After the window, nodes switch to their agreed upon channel and exchange messages on that channel for the rest of the beacon interval. Simulation results show that MMAC can achieve results comparable with the MAC protocols that require more than one transceiver for each node.

Designs across the network layer and transport layer, specifically TCP, [11, 30] and designs across the MAC layer and transport layer, specifically TCP, have also been proposed [21].

Current versions of TCP treat all packet losses as being related to congestion, which is mostly true in a wired network. However, due to the unreliability of wireless channels, many packets may be corrupted and lost due to channel degradation. If the TCP source reacts to the packet loss as if congestion was the cause and, thus, decreases its sending rate, network throughput may be unnecessarily reduced. Thus, congestion control schemes, such as random early detection (RED) [20] and its variants, are not effective in a wireless environment. Many TCP enhancements are proposed to address the issues that are related specifically to wireless networks. Holland and Vaidya [30] propose an explicit link failure notification (ELFN) technique to improve TCP performance in a wireless environment. The goal of ELFN is to differentiate congestion losses from link failure losses so that TCP will not invoke congestion control when a packet is lost due to link failures. The network layer notifies TCP when there is a link failure. Upon receiving this notice, TCP disables congestion control mechanisms and enters a stand-by mode until the route has been restored. Simulation results show that the use of ELFN can significantly improve TCP performance in a wireless environment. A similar approach is undertaken by Chandran, *et al.* [11], in which the same feedback scheme is used. A TCP source enters a “snooze” state upon receiving a Route Failure Notification (RFN) packet. Once in the snooze state, the source completely stops transmission and freezes all of its timers, the sending window for packets, and other state variables. Meanwhile, it starts a route failure timer which corresponds to a worst-case route re-establishment time. The source would remain in this state until it is notified of the restoration of the route through a Route Re-establishment Notification (RRN) packet.

Due to the performance limitation of the IEEE 802.11 MAC layer, TCP may suffer from decreased throughput and increased packet loss if its sending window size grows beyond an optimal window value. Unfortunately, if no measure is taken, TCP typically grows its window size much larger than the optimal one. Two techniques, link RED and adaptive pacing, are proposed by Fu, *et al.* [21] to improve TCP performance over multihop wireless networks. Similar to RED at the routing layer, in Link RED (LRED), the link layer maintains the average number of retries for recent packet transmissions. The head-of-line (HOL) packet is dropped from the buffer or marked with a probability based on this average number of retries. By marking HOL packets, LRED allows TCP flows with early congestion notification (ECN) to adapt their offered load without losing any packets. The goal of adaptive pacing is to improve channel reuse by distributing traffic among intermediate nodes in a more balanced way. This way, interference among packets within the same TCP flow can be reduced.

To achieve a higher success rate for accessing data in a mobile ad hoc network, Chen, Shah, and Nahrstedt [12] propose to share information between the middleware layer and the network layer. For instance, the data accessibility service at the middleware layer predicts the event of group partitioning and makes data replication decisions by utilizing the node's future location and connectivity information from the routing layer. Meanwhile, the routing layer takes into account the middleware's priority assigned to data to differentiate and prioritize network-level packets for scheduling purposes.

System-profile adaptive techniques are summarized in Table 7.2.

QoS-driven Cross-layer Designs

Because distinct applications may have different QoS requirements, network layers underneath the application layer need to co-operate with the application layer to provide appropriate QoS support and to ensure robustness and efficiency of data transmissions. Even though many QoS management strategies have been proposed for wired networks, they may not be suitable for use in a wireless network due to the challenges posed by the wireless environment. For instance, due to the inherent unreliability of a wireless link, it is impossible to guarantee at any time the fulfillment of QoS requirements at the physical layer. Under this

Table 7.2: An Overview of System-Adaptive Design Techniques

Ref.	Description	Goal
[31]	MAC rate adaptation based on channel information	Increase WLAN throughput and packet delivery ratio
[76, 65]	Distributed PHY channel assignment by MAC	Increase aggregate throughput
[30, 11]	Disable TCP congestion control upon receiving ELFN	Improve TCP performance in wireless environments
[21]	Link RED and adaptive pacing	Improve TCP performance in multihop wireless networks
[12]	Share information between routing and middleware layer	Achieve higher success rate for accessing data

condition, adaptation mechanisms need to be implemented at the data link layer or, possibly, also at higher layers to reduce the impact of an unreliable physical layer on QoS as much as possible. In spite of these mechanisms, general QoS support in a multi-hop ad hoc network is difficult, if not impossible, especially if nodes are mobile or the wireless environment is highly dynamic. The most current proposed approaches for cross-layer QoS are based on an adaptive QoS model, which requires applications to adapt to the time-varying resources provided by the network.

Some QoS-driven adaptive schemes are designed for one-hop wireless networks, where a central control node such as an access point is available. Other QoS-driven adaptive schemes are designed for wireless ad hoc networks, where nodes co-ordinate to achieve desired QoS level.

One-hop wireless networks

Kumwilaisak, *et al.* [44] propose a cross-layer QoS mapping architecture that can be used for video delivery in wireless networks. To provide progressive, fine granularity, and scalable video transmission over time-varying and non-stationary wireless channels, MPEG-4 frames from different layers in one group-of-picture (GOP) are assigned to different priority classes. Then, an optimization problem is formulated to provide minimum expected distortion in one GOP, subject to two sets of constraints. Once the optimal mapping solution is found, an optimal video QoS adaptation algorithm is formulated. The goal again is to minimize the expected video distortion, given a set of QoS bounds and current available wireless channel rate. This set of QoS bounds is adaptively chosen to cope with the time-varying wireless link. However, it is worth pointing out that adapting QoS requirements based on physical link variations is only applicable for a single-hop wireless link. For a multi-hop wireless ad hoc network, QoS requirements at the application layer should not adapt only in accordance with physical link degradations, mainly because there are multiple hops between source and destination and these links vary independently. Nevertheless, QoS requirements may be varied according to other indicators from the network, such as congestion levels detected at the link layer or the network layer.

In many wireless networks, link layers do not support forward error correction (FEC). Meanwhile, frame error coding and re-transmissions at the application layer will introduce excessive delays for some applications because the size of application layer data packets is often much larger than that of link layer data packets. To address such problems and to adapt to different application requirements and channel conditions, Shan and Zakhor [61] proposed an adaptive scheme at the application layer to provide unequal FEC and different level of protection to different packets within a video packet bit stream. The basic idea is to provide application layer with link layer information, such as which radio link protocol (RLP) packet is corrupted and the size of RLP packets. Using such information, application layer packets can be decomposed into RLP packets at the application layer. Unequal FEC can then be applied across RLP packets within an application layer packet, instead of across several application layer packets. Moreover, because the application layer knows which RLP packet is corrupted, it can explicitly ask the sender to re-transmit that RLP packet, rather than the whole application packet. Because FEC and automatic retransmission request (ARQ) are implemented at application layer, while the granularity is at the RLP packet size, the approach in [61] combines the flexibility and programmability of application layer adaptation with the low delay and bandwidth efficiency associated with link layer techniques.

Because wireless channels are unreliable and wireless receivers are often heterogeneous, it is challenging to provide QoS in such wireless networks. Wu, Hou, and Zhang [75] present an adaptive framework that consists of three basic components: 1) scalable video representations, 2) network-aware end system, and 3) adaptive services. Using scalable video coding, a raw video sequence is compressed into three layers: a base layer and two enhancement layers. A scalable video source can choose to transmit one or more layers based on the bandwidth fluctuation of a wireless channel. Knowing the status of underlying network resources, a network-aware end system can adapt the video stream accordingly and, thus, improve the performance of the application. Adaptive QoS support is a technique to adapt video streams during periods of QoS fluctuation and handoffs. Due to its channel-awareness and adaptivity, this framework can provide suitable QoS for video over wireless, while still achieving fairness in sharing resources.

Van der Schaar, *et al.* [73] propose a cross-layer error control and adaptation scheme for enhancing the robustness and efficiency of scalable video transmissions over 802.11 WLANs. To maximize video quality under different multipath channel conditions, the cross-layer protection strategy is developed to dynamically change three parameters: 1) the application layer frame error rate (FER), 2) the maximum MAC retransmission limit, and 3) the packet size. At the application layer, Reed-Solomon (RS) coding is applied across multiple video packets using an interleaver. By using different RS codes, throughput and delay can be traded off for stronger error protection. At the MAC layer, different MAC retransmission limits can be employed to trade off error protection with delay constraints. Moreover, optimal adaptive packet size selection is used to maximize the throughput efficiency for a given RS code and retransmission limit.

Table 7.3 summarizes QoS-driven adaptive techniques for one-hop wireless networks.

Table 7.3: An Overview of QoS-driven Techniques in a One-hop Wireless Environment

Ref.	Description	Goal
[44]	QoS mapping architecture	Minimize video distortion given available wireless channel rate
[61]	Adaptive and differentiated protections to packets	Achieve both flexibility and bandwidth efficiency
[75]	Adaptive QoS framework	Provide suitable QoS while achieving fair resource sharing
[73]	Error-control and adaptation scheme	Enhance robustness and efficiency of scalable video transmission

Multi-hop wireless ad hoc networks

While it is challenging to provide QoS in a wireless network, multi-hop wireless ad hoc networks pose even greater challenges. In a purely distributed ad hoc network, each node acts as both a source and a router. Distributed resource management and distributed packet scheduling have to be implemented. Ad hoc routing algorithms also greatly impact network throughput and video quality. Many multimedia transmission techniques utilize multipath routing algorithms to provide additional data throughput and error resilience. For instance, three video transport schemes are examined by Mao, *et al.* [51] to show the effectiveness of path diversity in combating transmission errors in ad hoc networks. The three schemes are: 1) feedback-based reference picture selection, 2) layered coding with selective automatic repeat request, and 3) multiple description motion compensated coding. All three schemes are based on close interactions between a multistream video codec at the application layer and the transport layer.

Distributed scheduling schemes in wireless ad hoc networks are investigated by Kanodia, *et al.* [39]. Two mechanisms for QoS communication in multi-hop ad hoc networks are proposed, namely a distributed priority scheduling scheme and a multi-hop coordination scheduling scheme. The distributed priority scheduling scheme piggybacks the priority tag of a node's HOL packet onto handshake and data packets, e.g. RTS/DATA in the IEEE 802.11 MAC. By monitoring transmitted packets, each node maintains a scheduling table that is used to assess the node's priority level relative to its neighboring nodes. The scheduling table is then incorporated into the existing IEEE 802.11 priority backoff schemes to approximate the idealized schedule. Recognizing the fact that multiple nodes in an ad hoc network need to coordinate to meet end-to-end QoS objectives, the authors also devise a multihop coordination scheduling scheme so that downstream nodes can increase a packet's relative priority to make up for excessive delays incurred upstream.

Setton, *et al.* [60] explore a cross-layer framework for real-time video streaming. In the framework, a general layered structure is maintained, but key parameters can be exchanged between adjacent layers. Within this context, adaptive link layer techniques adjust packet size, symbol rate, and modulation types according to varying channel conditions to improve physical layer throughput. At the MAC and network layers, the joint allocation of capacity and flow optimizes the supportable traffic rate significantly. Consequently, the end-to-end video quality can be improved. Smart scheduling at the transport layer further protects the video stream from packet losses and ensures timely arrivals of the video packets without causing excessive network congestion.

Li and Fang [48] propose a piggybacking scheme to support differentiated services in multi-hop wireless networks. Exploiting channel dynamics and stochastic traffic features, the piggybacking scheme lets high priority traffic help low priority traffic by sharing unused residual bandwidth with this traffic. Therefore, end-to-end delay and the packet delivery ratio can be improved. In this scheme, the MAC layer utilizes both packet priority information provided by the network layer and channel dynamics provided by the physical layer.

QoS-driven adaptive techniques generally take both the system profile and QoS requirements into consideration to ensure robust and efficient delivery of voice and video packets. The QoS-driven adaptive techniques for multi-hop wireless ad hoc networks are summarized

Table 7.4: An Overview of QoS-driven Techniques for Wireless Ad Hoc Networks

Ref.	Description	Goal
[51]	Video transport via multipath	Combat transmission errors via path diversity
[39]	Distributed scheduling	Meet end-to-end QoS objectives
[48]	Resource sharing by "piggybacking"	Improve end-to-end delay and packet delivery ratio

in Table 7.4.

7.3 Summary

In this chapter, an architecture for cross-layer design in wireless networks is proposed. Current cross-layer design research can be categorized into two classes: joint-layer design using optimization technique, and adaptive techniques based on a system profile and QoS requirements. Different types of network infrastructure, such as cellular networks, multi-hop wireless ad hoc networks, and sensor networks, have diverse requirements and pose unique challenges which need to be carefully considered in a cross-layer design. In other words, a good cross-layer design scheme needs to consider both the network infrastructure and interactions among layers.

Joint-layer design based on optimization techniques can achieve optimal performance, but at the expense of complexity. Adaptive schemes may achieve relatively good performance with less complexity. However, without careful design and a holistic view of the network architecture, adaptive schemes may backfire and cause more damage than benefit [41].

In an effort to improve the performance of wireless networks, a number of cross-layer techniques that rely on interactions between different layers have been proposed in the literature. However, many areas that may benefit from cross-layer techniques remain unexplored.

Moreover, cross-layer design can not only improve system performance, but may also reduce the complexity of the algorithms and protocols. As the proposed cross-layer based channel assignment algorithms, CA-AODV and CA-OLSR, have shown, problems that are difficult in a strict layered paradigm may become possible to solve if multiple layers can be jointly designed and optimized.

Chapter 8

Summary and Conclusions

8.1 Summary of the Dissertation

In this dissertation, I first presented three design principles for distributed channel assignment in wireless ad hoc networks. The first and primary principle is a cross-layer design principle that combines routing with channel assignment. The second design principle states that channels should be assigned on-demand to only active nodes. The third design principle states that channel assignment should consider both collisions and interference.

Based on these design principles, I investigated two example protocols, CA-AODV and CA-OLSR. CA-AODV also has two extensions: E2-CA-AODV and Ek-CA-AODV. These channel assignment protocols exhibit significantly lower communication, computation, and storage complexity than existing channel assignment schemes. In addition, they require fewer channels than many existing channel assignment algorithms.

I proved the correctness of the proposed algorithms and derived an upper bound on the number of channels required to resolve collisions as well as to mitigate mutual interference. Simulation results show that, in some cases, the performance of CA-AODV and CA-OLSR can approach that of a scheme that utilizes an unlimited number of channels. Further, a numerical example shows that Ek-CA-AODV can give channel assignments similar to the

Greedy-AODV algorithm in some cases.

Secondly, a new multi-channel medium access control (MC-MAC) protocol was proposed for multi-hop wireless ad hoc networks. MC-MAC has two unique design features compared with the traditional IEEE 802.11 MAC protocol. First, to ensure maximum fairness in the network, MC-MAC does not utilize a binary exponential backoff scheme. Secondly, MC-MAC utilizes a modified virtual carrier sensing scheme to mitigate the exposed node problem and to improve performance. As far as I know, MC-MAC is the first multi-channel MAC protocol that explicitly takes both the primary design goal, improving the throughput, and a secondary design goal, fairness, into consideration. Additionally, MC-MAC is compatible with the IEEE 802.11 MAC layer and imposes the minimum system requirements among all existing multi-channel MAC protocols. Through a simulation study and capacity analysis, it was shown that MC-MAC, by effectively increasing capacity, substantially increases throughput and decreases delay compared to the single-channel IEEE 802.11 MAC protocol. Therefore, MC-MAC and the accompanying distributed channel assignment protocols constitute an effective solution to the performance problem in a multi-hop wireless network.

Finally, the cross-layer design principle was generalized to other networking problems and a cross-layer design architecture was proposed. The proposed architecture supports adaptation and optimization both at a single layer and across multiple layers if necessary. This cross-layer design architecture not only classifies existing cross-layer design techniques, but also facilitates future cross-layer designs. A survey of existing cross-layer design techniques was presented to validate the architecture.

8.2 Summary of Contributions

This dissertation has three major contributions. The primary contribution is the proposal and design of example protocols based on the three design principles for distributed channel assignment problem. The protocols that utilize the proposed design principles are shown to require fewer channels and exhibit significantly lower communication, computation, and storage complexity than existing approaches. Traditionally, distributed channel assignment and ad hoc routing are two different research topics. However, under the cross-layer design principle, these two research areas can be combined together to provide more efficient protocols that have superior performance. The other two design principles also contribute to the effectiveness of the proposed channel assignment protocols.

The second contribution is a new multi-channel MAC (MC-MAC) protocol. The proposed MC-MAC is the first multi-channel MAC protocol that takes a secondary design objective, such as fairness, into consideration. In addition, MC-MAC is compatible with the IEEE 802.11 MAC and has the lowest system requirement among all existing multi-channel MAC protocols.

The third contribution is a unique cross-layer design architecture. Realizing the potential benefits of cross-layer design, I propose a cross-layer design architecture as a unifying framework to guide future cross-layer designs. The proposed architecture maintains two global data structures that facilitate efficient information exchange among different layers. A literature review of the current state-of-the-art validates this architecture.

8.3 Future Research Directions

This dissertation focuses on the improving capacity of wireless ad hoc networks when each host is equipped with a single half-duplex transceiver, i.e. one NIC. Because of this hardware constraint, the benefits of distributed channel assignment may not be fully explored. In the future, if each node can be equipped with multiple transceivers, a new combined channel assignment and multi-channel MAC protocol may yield even more significant performance improvement than the proposed MC-MAC protocol. This new multi-channel MAC protocol might utilize multiple channels more efficiently due to the multiple available transceivers. In addition, the new protocol should be able to mitigate hidden node, exposed node, and deafness problems.

The second research direction would be to design a multi-channel MAC protocol that can co-exist with the IEEE 802.11 MAC protocol. The proposed MC-MAC protocol does not have a binary exponential backoff mechanism. Thus, it is more aggressive than the IEEE 802.11 MAC protocol when contending for wireless media. A new multi-channel MAC protocol that is fully compatible with the IEEE 802.11 MAC but still provides multi-channel capacity improvement is ideal.

Another research direction is to apply the proposed design principles to other ad hoc routing protocols, such as TBRPF [53]. Even though all routing protocols may have the same design goal, i.e. providing efficient and accurate routing, they might be very different in terms of their implementation and other details. The design of distributed channel assignment protocols should introduce minimum changes to the routing protocols, but still achieve the desired performance and efficiency.

The ultimate research direction is to extend and to apply the proposed cross-layer design architecture. As a unifying framework, the cross-layer design architecture serves as a starting point for many networking problems that include not only theoretical work, but also real-world applications. Currently, there are few systems, even prototypes, that actually implement cross-layer design techniques. In the future, more systems could be designed and implemented to test the performance of cross-layer techniques.

Appendix A

List of Acronyms

ACK	Acknowledgement
AODV	Ad hoc On Demand Distance Vector routing
ARQ	Automatic Repeat reQuest
AWGN	Additive White Gaussian Noise
BEB	Binary Exponential Backoff
CA-AODV	Channel Assignment-AODV
CA-OLSR	Channel Assignment-Optimized Link State Routing
CCA	Common Code Assignment
CDMA	Code Division Multiple Access
CSMA	Carrier Sense Multiple Access
CSMA/CA	CSMA with Collision Avoidance
CTS	Clear To Send
DCA	Dynamic Channel Assigment
DCF	Distributed Coordination Function
DIFS	DCF InterFrame Space
DSDV	Destination-Sequenced Distance Vector

E2-CA-AODV	Enhanced 2-hop CA-AODV
Ek-CA-AODV	Enhanced k-hop CA-AODV
ELFN	Explicit Link Failure Notification
FAMA	Floor Acquisition Multiple Access protocol
FDMA	Frequency Division Multiple Access
FEC	Forward Error Correction
FER	Frame Error Rate
IP	Internet Protocol
MAC	Medium Access Control
MACA-P	Medium Access via Collision Avoidance with enhanced Parallelism
MANET	Mobile Ad Hoc NETwork
MC-MAC	Multi-Channel MAC protocol
MILD	Multiplicative Increase and Linear Decrease
MPR	Multi-Point Relay
MMAC	Multi-channel MAC
MSE	Mean Squared Error
NAV	Network Allocation Vector
NIC	Network Interface Card
OLSR	Optimized Link State Routing
PCA	Pair-wise Code Assignment
QoS	Quality of Service
RCA	Receiver-based Code Assignment
RFC	Request For Comment
RFN	Route Failure Notification
RLP	Radio Link Protocol
RICH-DP	Receiver-Initiated Channel Hopping with Dual Polling
RREQ	Route Request message
RREP	Route Reply message

RTS	Request To Send
SIFS	Short InterFrame Space
SINR	Signal-to-Interference-Noise Ratio
SSCH	Slotted Seeded Channel Hopping
TBRPF	Topology Dissemination Based on Reverse-Path Forwarding
TC	Topology Control
TCA	Transmitter-based Code Assignment
TCP	Transport Control Protocol
TDMA	Time Division Multiple Access
TTL	Time-To-Live
UDP	User Datagram Protocol
WLAN	Wireless Local Area Network

Bibliography

[1] "The Cocoon and the Butterfly," *www.journeyofhearts.org/jofh/kirstimd/bflycocoon.htm.*

[2] *IEEE Std. 802.11, Wireless LAN media access control (MAC) and physical layer (PHY) specifications*, 1997.

[3] N. Abramson, "The ALOHA system - another alternative for computer communications," in *AFIPS Fall Joint Computer Conf.*, vol. 37, 1970, pp. 281–285.

[4] A. Acharya, A. Misra, and S. Bansal, "MACA-P: A MAC for concurrent transmissions in multi-hop wireless networks," in *Proc. IEEE International Conference on Pervasive Computing and Communications (PerCom)*, 2003, p. 505.

[5] P. Bahl, R. Chandra, and J. Dunagan, "SSCH: Slotted seeded channel hopping for capacity improvement in IEEE 802.11 ad-hoc wireless networks," in *Proc. IEEE/ACM the Tenth Annual International Conference on Mobile Computing and Networking (MobiCom)*, 2004, pp. 216–230.

[6] A. Bertossi and M. Bonuccelli, "Code assignments for hidden terminal interference avoidance in multihop packet radio networks," *IEEE/ACM Trans. on Networking*, vol. 3, no. 4, pp. 441–449, Aug. 1995.

[7] D. Bertsekas and R. Gallager, *Data Networks.* Upper Saddle River, NJ: Prentice Hall, 1992.

[8] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, "MACAW: A media access protocol for wireless LANs," in *Proc. SIGCOMM*, 1994, pp. 212–225.

[9] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 3, pp. 535–547, March 2000.

[10] G. Bianchi and I. Tinnirello, "Kalman filter estimation of the number of competing terminals in an IEEE 802.11 network," in *Proc. IEEE INFOCOM*, 2003, pp. 844–852.

[11] K. Chandran, S. Raghunathan, S. Venkatesan, and R. Prakash, "A feedback-based scheme for improving TCP performance in ad hoc wireless networks," *IEEE Personal Communications Magazine*, vol. 8, no. 1, pp. 34–39, Feb. 2001.

[12] K. Chen, S. H. Shah, and K. Nahrstedt, "Cross-layer design for data accessibility in mobile ad hoc networks," *Journal of Wireless Personal Communications*, vol. 21, pp. 49–76, April 2002.

[13] M. Chiang, "To layer or not to layer: Balancing transport and physical layer in wireless multihop networks," in *Proc. INFOCOM*, 2004, pp. 2525–2536.

[14] T. Clausen and P. J. eds., "Optimized Link State Routing Protocol," Oct. 2003, IETF RFC 3626.

[15] M. Conti, G. Maselli, G. Turi, and S. Giordano, "Cross-layering in mobile ad hoc network design," *IEEE Computer*, vol. 37, no. 2, pp. 48–51, Feb. 2004.

[16] R. Cruz and A. Santhanam, "Optimal routing, link scheduling and power control in multi-hop wireless networks," in *Proc. INFOCOM*, 2003, pp. 702–711.

[17] J. D. Day and H. Zimmermann, "The OSI reference model," *Proc. of IEEE*, vol. 71, no. 12, pp. 1334–1340, Dec. 1983.

[18] T. Elbatt and A. Ephremides, “Joint scheduling and power control for wireless ad-hoc networks,” *IEEE Trans. on Wireless Communications*, vol. 3, no. 1, pp. 74–85, Jan. 2004.

[19] Z. Fang, B. Bensaou, and Y. Wang, “Performance evaluation of a fair backoff algorithm for IEEE 802.11 DFWMAC,” in *Proc. 3rd ACM International Symposium on Mobile Ad hoc Networking and Computing (MOBIHOC)*, 2002, pp. 48–57.

[20] S. Floyd and V. Jacobson, “Random early detection gateways for congestion avoidance,” *IEEE/ACM Trans. on Networking*, vol. 1, no. 4, pp. 397–413, Aug. 1993.

[21] Z. Fu, P. Zerfos, H. Luo, S. Lu, L. Zhang, and M. Gerla, “The impact of multihop wireless channel on TCP throughput and loss,” in *Proc. INFOCOM*, vol. 3, 2003, pp. 1744–1753.

[22] C. L. Fullmer and J. J. Garcia-Luna-Aceves, “Floor acquisition multiple access (FAMA) for packet-radio networks,” in *Proc. SIGCOMM*, 1995, pp. 262–273.

[23] J. J. Garcia-Luna-Aceves and J. Raju, “Distributed assignment of codes for multihop packet-radio networks,” in *Proc. MILCOM*, vol. 1, 1997, pp. 450–454.

[24] A. J. Goldsmith and S. B. Wicker, “Design challenges for energy-constrained ad hoc wireless networks,” *IEEE Wireless Communications*, vol. 9, no. 4, pp. 8–27, Aug. 2002.

[25] M. X. Gong and S. F. Midkiff, “Distributed channel assignment protocols: a cross-layer approach,” in *Proc. IEEE Wireless Communication and Networking Conference (WCNC)*, 2005, pp. 2195–2200.

[26] ——, “A new multi-channel MAC protocol based a cross-layer approach,” 2005, submitted for publication.

[27] M. X. Gong, S. F. Midkiff, and S. Mao, “A combined proactive routing and multi-channel MAC protocol for wireless ad hoc networks,” in *Proc. IEEE BROADNETS, invited paper*, 2005.

[28] ——, “Design principles for distributed channel assignment in wireless ad hoc networks,” in *Proc. IEEE International Communications Conference (ICC)*, 2005.

[29] P. Gupta and P. R. Kumar, “The capacity of wireless networks,” *IEEE Trans. on Information Theory*, vol. 46, no. 2, pp. 388–404, March 2000.

[30] G. Holland and N. Vaidya, “Analysis of TCP performance over mobile ad hoc networks,” in *Proc. IEEE/ACM the International Conference on Mobile Computing and Networking (MobiCom)*, 1999, pp. 219–230.

[31] G. Holland, N. Vaidya, and P. Bahl, “A rate-adaptive MAC protocol for multi-hop wireless networks,” in *Proc. IEEE/ACM the International Conference on Mobile Computing and Networking (MobiCom)*, 2001, pp. 236–251.

[32] L. Hu, “Distributed code assignments for CDMA packet radio networks,” *IEEE/ACM Trans. on Networking*, vol. 1, no. 6, pp. 668–677, Dec. 1993.

[33] I. S. Institute, “Transmission control protocol,” Sept. 1981, IETF RFC 793.

[34] K. Jain, J. Padhye, V. Padmanabhan, and L. Qiu, “Impact of interference on multi-hop wireless network performance,” in *Proc. IEEE/ACM the International Conference on Mobile Computing and Networking MobiCom*, 2003, pp. 66–80.

[35] N. Jain, S. R. Das, and A. Nasipuri, “A multichannel CSMA MAC protocol with receiver-based channel selection for multihop wireless networks,” in *Proc. IEEE International Conference on Computer Communications and Networks (IC3N)*, 2001, pp. 432–439.

[36] M. Joa-Ng and I.-T. Lu, “Spread spectrum medium access protocol with collision avoidance in mobile ad-hoc wireless network,” in *Proc. INFOCOM*, vol. 2, 1999, pp. 21–25.

[37] M. Johansson, L. Xiao, and S. Boyd, “Simultaneous routing and resource allocation in CDMA wireless data networks,” in *Proc. IEEE International Communications Conference (ICC)*, vol. 1, 2003, pp. 51–55.

[38] D. Johnson, D. Maltz, and J. Broch, *DSR: The Dynamic Source Routing Protocol for Multihop Wireless Ad Hoc Networks.* Addison-Wesley, 2001, ch. 5, pp. 139–172.

[39] V. Kanodia, C. Li, A. Sabharwal, B. Sadeghi, and E. Knightly, "Distributed priority scheduling and medium access in ad hoc networks," *Wireless Networks*, vol. 8, pp. 455–466, Sept. 2002.

[40] P. Karn, "MACA: A new channel access protocol for packet radios," in *Proc. ARRL/CRRL Amateur Radio 9th Comp. Net. Conf.*, 1990, pp. 134–140.

[41] V. Kawadia and P. R. Kumar, "A cautionary perspective on cross layer design," *IEEE Wireless Communications*, vol. 12, no. 1, pp. 3–11, Feb. 2005.

[42] L. Kleinrock and F. Tobagi, "Packet switching in radio channels: Part I - carrier sense multiple-access modes and their throughput-delay characteristics," *IEEE Trans. on Communications*, vol. COM-23, no. 12, pp. 1400–1416, Dec. 1975.

[43] U. Kozat, I. Koutsopoulos, and L. Tassiulas, "A framework for cross-layer design of energy-efficient communication with QoS provisioning in multi-hop wireless networks," in *Proc. INFOCOM*, 2004, pp. 1446–1456.

[44] W. Kumwilaisak, Y. T. Hou, Q. Zhang, W. Zhu, C.-C. Kuo, and Y.-Q. Zhang, "A cross-layer quality-of-service mapping architecture for video delivery in wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 21, no. 10, pp. 1685–1698, Dec. 2003.

[45] P. Kyasanur and N. Vaidya, "Capacity of multi-channel wireless networks: impact of number of channels and interfaces," University of Illinois at Urbana-Champaign, Tech. Rep., 2005.

[46] ——, "Routing and interface assignment in multi-channel multi-interface wireless networks," in *Proc. IEEE Wireless Communications and Networking Conference (WCNC)*, 2005.

[47] J. Li, C. Blake, D. S. J. D. Coute, H. I. Lee, and R. Morris, “Capacity of ad hoc wireless networks,” in *Proc. IEEE/ACM International Conference on Mobile Computing and Networking (MobiCom)*, 2001, pp. 61–69.

[48] W. Liu and Y. Fang, “Courtesy piggybacking: Supporting differentiated services in multihop mobile ad hoc networks,” in *Proc. INFOCOM*, 2004.

[49] X. Lu, E. Erkip, Y. Wang, and D. Goodman, “Power efficient multimedia communication over wireless channels,” *IEEE Journal on Selected Areas in Communications*, vol. 21, no. 10, pp. 1738–1750, Dec. 2003.

[50] A. Maharshi, L. Tong, and A. Swami, “Cross-layer designs of multichannel reservation MAC under Rayleigh fading,” *IEEE Trans. on Signal Processing*, vol. 51, no. 8, pp. 2054–2067, Aug. 2003.

[51] S. Mao, S. Lin, S. S. Panwar, Y. Wang, and E. Celebi, “Video transport over ad hoc networks: Multistream coding with multipath transport,” *IEEE Journal on Selected Areas in Communications*, vol. 21, no. 10, pp. 1721–1736, Dec. 2003.

[52] T. Nandagopal, T. Kim, X. Gao, and V. Bharghavan, “Achieving MAC layer fairness in wireless packet networks,” in *Proc. IEEE/ACM International Conference on Mobile Computing and Networking (MobiCom)*, 2000, pp. 87–98.

[53] R. Ogier, F. Templin, and M. Lewis, “Topology Dissemination Based on Reverse-Path Forwarding (TBRPF),” Feb. 2004, IETF RFC 3684.

[54] T. Ozugur, M. Naghshineh, P. Kermani, and J. A. Copeland, “Fair media access for wireless LANs,” in *Proc. IEEE GLOBECOM*, 1999, pp. 570–579.

[55] C. Perkins, E. Belding-Royer, and S. Das, “Ad hoc On-Demand Distance Vector (AODV) routing,” July 2003, IETF RFC 3661.

[56] C. E. Perkins and P. Bhagwat, “Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers,” in *Proc. ACM SIGCOM*, 1994, pp. 234–244.

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

[58] Q. Qiao and S. Choi, "Goodput enhancement of IEEE 802.11a wireless LAN via link adaptation," in *Proc. IEEE International Communications Conference (ICC)*, 2001, pp. 1995–2000.

[59] A. Raniwala and T. Chiueh, "Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network," in *Proc. IEEE INFOCOM*, 2005.

[60] E. Setton, T. Yoo, X. Zhu, A. J. Goldsmith, and B. Girod, "Cross-layer design of ad hoc networks for real-time video streaming," *IEEE Wireless Communications*, 2005, to appear.

[61] Y. Shan and A. Zakhor, "Cross layer techniques for adaptive video streaming over wireless networks," in *Proc. International Conference on Multimedia and Expo*, 2002, pp. 277–280.

[62] D. Shukla, L. Chandran-Wadia, and S. Iyer, "Mitigating the exposed node problem in IEEE 802.11 ad hoc networks," in *Proc. the 12th International Conference on Computer Communications and Networks (ICCCN)*, 2003, pp. 157–162.

[63] J. So and N. Vaidya, "Multi-channel MAC for ad hoc networks: Handling multi-channel hidden terminals using a single transceiver," in *Proc. ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, 2004, pp. 222–233.

[64] ——, "Routing and channel assignment in multi-channel multi-hop wireless networks with single-NIC devices," University of Illinois at Urbana-Champaign, Tech. Rep., 2005.

[65] ——, "Multi-channel MAC for ad hoc networks: Handling multi-channel hidden terminals using a single transceiver," in *Proc. ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, May 2004.

[66] M. Soroushnejad and E. Geraniotis, "Probability of capture and rejection of primary multiple-access interference in spread-spectrum networks," *IEEE Trans. on Communications*, vol. 39, no. 6, pp. 986–994, June 1991.

[67] E. S. Sousa and J. A. Silvester, "Spreading code protocols for distributed spread-spectrum packet radio networks," *IEEE Trans. on Communications*, vol. 36, no. 3, pp. 272–281, March 1988.

[68] K. Stuhlmuller, N. Farber, M. Link, and B. Girod, "Analysis of video transmission over lossy channels," *IEEE Journal on Selected Areas in Communications*, vol. 18, pp. 20–31, June 2000.

[69] J. Tang, G. Xue, and W. Zhang, "Interference-aware topology control and QoS routing in multi-channel wireless mesh networks," in *Proc. the 6th ACM international symposium on Mobile ad hoc networking and computing (MobiHoc)*, May 2005, pp. 68–77.

[70] C. K. Toh, "Maximum battery life routing to support ubiquitous mobile computing in wireless ad hoc network," *IEEE Communications Magazine*, vol. 39, no. 6, pp. 2–11, June 2001.

[71] S. Toumpis and A. J. Goldsmith, "Performance, optimization, and cross-layer design of media access protocols for wireless ad hoc networks," in *Proc. IEEE International Conference on Communications (ICC)*, vol. 3, 2003, pp. 2234–2240.

[72] A. Tzamaloukas and J. J. Garcia-Luna-Aceves, "A receiver-initiated collision-avoidance protocol for multi-channel networks," in *Proc. IEEE INFOCOM*, 2001, pp. 189–198.

[73] M. van der Schaar, S. Krishnamachari, S. Choi, and X. Xu, "Adaptive cross-layer protection strategies for robust scalable video transmission over 802.11 WLANs," *IEEE Journal on Selected Areas in Communications*, vol. 21, no. 10, pp. 1752–1763, Dec, 2003.

[74] VINT Group, *UCB/LBNL/VINT network simulator ns (version 2).*

[75] D. Wu, Y. Hou, and Y. Zhang, “Scalable video coding and transport over broadband wireless networks,” *Proc. of the IEEE*, vol. 81, no. 1, pp. 6–20, 2001.

[76] S.-L. Wu, C.-Y. Lin, Y.-C. Tseng, and J.-P. Sheu, “A new multi-channel MAC protocol with on-demand channel assignment for multihop mobile ad hoc networks,” in *Proc. International Symposium on Parallel Architectures, Algorithms and Networks (ISPAN)*, 2000, pp. 232–237.

[77] L. Xiao, M. Johansson, and S. Boyd, “Simultaneous routing and resource allocation via dual decomposition,” *IEEE Trans. on Communications*, vol. 52, no. 7, pp. 1136–1144, July 2004.

[78] S. Xu and T. Saadawi, “Does the IEEE 802.11 MAC protocol work well in multihop wireless ad hoc networks?” *IEEE Communications Magazine*, vol. 39, no. 6, pp. 130–137, June 2001.

[79] X. Yang and N. H. Vaidya, “A wireless MAC protocol using implicit pipelining,” *IEEE Transactions on Mobile Computing*, to appear.

[80] L. Zhang, Y. Ge, and J. Hou, “Energy-efficient real-time scheduling in IEEE 802.11 wireless LANs,” in *Proc. International Conference on Distributed Computing Systems (ICDCS)*, 2003, pp. 628–635.

Vita

Michelle X. Gong received her B.S. degree in Electrical Engineering from Wuhan University, China, in 1996 and her M.S. degree in Electrical Engineering from University of Hawaii in 2000. She also studied in the masters program at Wuhan University from 1996 to 1998.

From 2000 to 2002, Michelle worked as a system engineer in LinkAir Communications, Inc., San Jose, California. Her main job responsibilities included designing MAC layer protocols and handoff mechanisms in a 3G cellular network. She has a pending US patent on handoff mechanisms in mobile wireless networks.

From 2002 and 2005, Michelle worked as a graduate research assistant and an NSF-funded IGERT Fellow in Bradley Department of Electrical and Computer Engineering, Virginia Tech, Virginia. Her current research interests are multi-channel utilization in mobile ad hoc networks, MAC layer design, ad hoc routing, and wireless mesh networks.

Publications

1. Book chapter: M. X. Gong and S. F. Midkiff, "*UWB networking*," in An Introduction to Ultra Wideband Communication Systems, Prentice Hall, 2005.
2. M. X. Gong, S. F. Midkiff, and S. Mao "A combined proactive routing and multi-channel MAC protocol for wireless ad hoc networks," in *Proc. IEEE BROADNETS, invited paper*, 2005.
3. M. X. Gong, S. F. Midkiff, and S. Mao, "Design principles for distributed channel

assignment in wireless ad hoc networks," in *Proc. IEEE International Conference on Communication (ICC)*, Seoul, Korea, 2005.

4. M. X. Gong and S. F. Midkiff, "Distributed channel assignment protocols: a cross-layer approach," in *Proc. IEEE Wireless Communications and Networking Conference (WCNC)*, New Orleans, USA, 2005.

5. M. X. Gong, S. F. Midkiff, and R. M. Buehrer, "A self-organized clustering algorithm for UWB ad hoc networks," in *Proc. IEEE Wireless Communications and Networking Conference (WCNC)*, Atlanta, USA, March 2004.

6. M. X. Gong and S. F. Midkiff, "Medium access control schemes for UWB ad hoc networks," in *The Ultra Wide Band Summit*, Paris, France, Dec. 2003.

7. M. X. Gong, S. F. Midkiff, and R. M. Buehrer, "A new piconet formation protocol for UWB ad-hoc networks," in *Proc. IEEE Conf. on Ultra Wideband Systems and Technologies (UWBST)*, Reston, USA, Nov. 2003.

8. A. Kuh and X. Gong, "Independent component analysis for blind multi-user detection," in *Proc. IEEE International Symposium on Information Theory*, Sorrento, Italy, June 2000.

9. A. Kuh and X. Gong, "Optimal margin classifier for multi-user detection in wireless communications," in *Proc. 1999 International Symposium on Nonlinear Theory and Its Applications*, Hawaii, Nov., 1999.

10. X. Gong and A. Kuh, "Support vector machine for multi-user detection in code-division multiple access communications," in *Proc. 33rd Asilomar Conference on Signals, Systems and Computers*, California, 1999.

11. N. Ma and X. Gong, "A multi-resolution reconstruction algorithm based on network theory for electrical capacitance tomography," *Wuhan University Journal of Natural Science*, vol. 3, no. 1, March 1998.