# ABSTRACT


Title of dissertation: ENABLING DYNAMIC SPECTRUM ALLOCATION IN COGNITIVE RADIO NETWORKS

Yuan Yuan, Doctor of Philosophy, 2007

Dissertation directed by: Associate Professor William A. Arbaugh, Department of Computer Science



The last decade has witnessed the proliferation of innovative wireless technologies, such as Wi-Fi,wireless mesh networks, operating in unlicensed bands. Due to the increasing popularity and the wide deployments of these technologies, the unlicensed bands become overcrowded. The wireless devices operating in these bands interfere with each other and hurt the overall performance. To support fast growths of wireless technologies, more spectrums are required. However, as most "prime" spectrum has been allocated, there is no spectrum available to expand these innovative wireless services. Despite the general perception that there is an actual spectral shortage, the recent measurement results released by the FCC (Federal Communications Commission) show that on average only 5% of the spectrum from 30MHz to 30 GHz is used in the US. This indicates that the inefficient spectrum usage is the root cause of the spectral shortage problem. Therefore, this dissertation is focused on improving spectrum utilization and efficiency in tackling the spectral shortage problem to support ever-growing user demands for wireless applications.

This dissertation proposes a novel concept of dynamic spectrum allocation, which adaptively divides available spectrum into non-overlapping frequency seg-

ments of different bandwidth considering the number of potentially interfering transmissions and the distribution of traffic load in a local environment. The goals are (1) to maximize spectrum efficiency by increasing parallel transmissions and reducing co-channel interferences, and (2) to improve fairness across a network by balancing spectrum assignments. Since existing radio systems offer very limited flexibility, cognitive radios, which can sense and adapt to radio environments, are exploited to support such a dynamic concept.

We explore two directions to improve spectrum efficiency by adopting the proposed dynamic allocation concept. First, we build a cognitive wireless system called KNOWS to exploit unoccupied frequencies in the licensed TV bands. KNOWS is a hardware-software platform that includes new radio hardware, a spectrum-aware MAC (medium access control) protocol and an algorithm for implementing the dynamic spectrum allocation. We show that KNOWS accomplishes a remarkable 200% throughput gain over systems based on fixed allocations in common cases. Second, we enhance Wireless LANs (WLANs), the most popular network setting in unlicensed bands, by proposing a dynamic channelization structure and a scalable MAC design. Through analysis and extensive simulations, we show that the new channelization structure and the scalable MAC design improve not only network capacity but per-client fairness by allocating channels of variable width for access points in a WLAN.

As a conclusion, we believe that our proposed concept of dynamic spectrum allocation lays down a solid foundation for building systems to efficiently use the invaluable spectrum resource.

# ENABLING DYNAMIC SPECTRUM ALLOCATION IN COGNITIVE RADIO NETWORKS

by

Yuan Yuan

Dissertation submitted to the Faculty of the Graduate School of
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2007

Advisory Committee:

Associate Professor William Arbaugh, Chairman/Advisor
Professor Agrawala Ashok
Professor Raymond Miller
Dr. Victor Bahl
Professor Mark Shayman, Dean's Representative

# DEDICATION

To my Father who loves me so deeply and
gives me everything I need to explore the world
To my Mother who is so kind and beautiful
and who is the only reason for me to stay strong

To my husband whose love has enriched
my life with happiness, peace, and hope
To my brother who always inspires me
and whom I love wholeheartedly

# ACKNOWLEDGEMENTS

It is a pleasure to express my deep gratitude to my advisors who make it possible to complete this thesis. I am greatly indebted to my advisor Prof. William Arbaugh, whose guidance, stimulating suggestions and encouragements helped me in all the time of my research. Much of my success can be attributed to the numerous doors he opened for me and his guidance over the last six years. I am genuinely thankful to my co-advisor Prof. Agrawala Ashok, who constantly fueled me with strength, hope and support to take on challenges and to go through frustrations. I am much obligated to Prof. Songwu Lu, who was cordially persistent in steering me and instructing me to explore more deeply in research. With great gratitude, I thank Dr. Victor Bahl, who believed in me, pushed me, and directed me, and now I have accomplished far more than I would have thought possible one year ago.

I would like to give my special thanks to the committee members. Especially, Professor Raymond Miller patiently listened to me; and his insightful and hearty advices greatly helped to further improve quality of this dissertation. Professor Mark A. Shayman provided detail suggestions and comments in guiding me to further improve consistency and accuracy of the presentation.

I would also like to thank researchers whom I have worked with. Dr. Daqing Gu and Dr. Jinyun Zhang gave me the first opportunity to work on wireless network technologies in research labs. I have learned a lot from discussions with the researchers in Cognitive Radio & Networking Group of Microsoft Research, Redmond. They are Dr. Ranveer Chandra, Dr. Philip A. Chou, Ian Ferrel, Dr. Amer Hassan, Dr. Thomas Moscibroda, Srihari Narlanka, and Dr. Yunnan Wu. Dr. Moscibroda theocratically formulated the problem of dynamic spectrum allocation and analyzed the KNOWS system. Some parts of Section 3.4 and Section 4.1

are based on his contributions.

Finally, I owe a debt of gratitude to my family members for their unconditional love, patience and understanding.

# Contents

# List of Tables

# List of Figures

# Chapter 1

# Introduction

*The significant problems we face*
*cannot be solved at the same level of thinking*
*we were at when we created them.*
*— Albert Einstein*
*To raise new questions, new possibilities,*
*to regard old problems from a new angle,*
*requires creative imagination*
*and marks real advance in science.*
*— Albert Einstein*

In the last decade, the small segment of unlicensed spectrum, i.e., the 2.4G ISM band, has spurred the proliferation of innovative wireless technologies including Wi-Fi and wireless mesh networks. The wide adoption of these technologies is then accelerated by increasing user demands for wireless access, inexpensive wireless equipments, and broader Internet access availability. Consequently, the unlicensed bands have become increasingly crowed in recent years, and thus the unlicensed wireless devices interfere with each other and hurt the overall performance. To support fast growths of these wireless technologies, more spec-

trum resources are required. However, at present time most "prime" spectrum has been assigned, and it is becoming increasingly difficult to find spectrum that can be made available either for new services or to expand existing ones. Despite the general perception that there is a real spectral shortage, the measurement results released by the Federal Communications Commission (FCC) indicate that for 90% of the time many licensed frequency bands remain unused [33]. This suggests that it is not an actual spectral shortage that is worrisome, but rather the inefficient spectrum usage. As demands on the number of users and their data rates steadily increase, it is important to enable efficient access to and use of the radio spectrum.

The dissertation is focused on improving spectrum utilization and efficiency in tackling this "artificial" spectral shortage problem. Without requiring any new allocation of spectrum, the solution holds a great potential to support growths of innovative wireless technologies. The deployment of these wireless technologies is unencumbered by regulatory delay, and thus will result in a plethora of new applications including last-mile broadband wireless access, health care, wireless PANs/LANs/MANs.

Our proposed solution for improving spectrum utilization and efficiency addresses the spectral shortage problem from the following two perspectives:

- The first part of solution makes use of idle frequencies in the licensed spectrum to improve spectrum utilization. This part of solution is motivated by the recent measurement results of radio frequencies and the regulation reformations. The recent measurements have shown that a large portion of the licensed bands is extremely under-utilized. For example, the average utilization of the licensed spectrum for television (TV) broadcast was as low as 14% in 2004, and this number keeps decreasing every year [85]. To improve spectrum utilization, the FCC is proposing the new rules that per-

mit the usage of unlicensed devices in the licensed TV bands when there is no licensed (primary) user operating on it [31]. These idle frequencies, so called "white spaces", are great resources for wireless data communications. In this dissertation, we focus on white spaces in the TV bands, which FCC will make available for unlicensed access in 2009 [74]. This sub-GHz spectrum has several properties that makes it desirable for data communications. In particular, radio frequency (RF) communications can occur over longer distances and RF waves have better penetration property in the lower frequency bands compared to the higher frequency ISM bands.

- The second part of the solution aims at improving spectrum efficiency in the crowed unlicensed bands. This part of solution is focused on Wireless Local Area Networks (WLANs), the most popular setup in the unlicensed spectrum. A WLAN is comprised of multiple Access Points (APs), each of which is responsible for delivering traffic from and to the wireless clients within its service area. The unlicensed spectrum is divided into small channels of equal bandwidth, and each AP operates on a particular channel. For example, the 2.4 GHz ISM band has 3 non-overlapping channels and an AP usually chooses the least congested channel to operate. The IEEE 802.11 Medium Access Control (MAC) regulates the channel access for competing wireless clients served by the same AP. However, the inefficiency of WLANs has been revealed by the recent measurements of user and traffic in real-world WLANs. The measurements show great variations in user and traffic in large-scale WLANs [23, 25, 95]. More specifically, certain APs become hotspot, serving a large number of users and a large amount of traffic, while others remain unused. And the set of heavily loaded APs changes over time. Consequently, the existing solution of using the fixed channels and IEEE 802.11 MAC has largely constrained capacity and fair-

ness of WLANs. To make it concrete, consider an example that an AP near a conference serving multiple clients obtains the same amount of spectrum, a channel, as the one in the corner serving very few users. As a result, the channel of equal bandwidth reduces the overall achievable capacity of the network; and the wireless clients receive unfair services depending on their locations. Moreover, the IEEE 802.11 MAC does not deal effectively with variations in user populations and physical-layer capabilities [101]. Therefore, while we are searching for new spectrum resources, it is equally important to make efficient use of the existing unlicensed bands and provide a scalable solution to serve different client sizes and various traffic loads.

In order to provide these two parts of the solution, we identify the key challenges for each part as elaborated below.

**Using While Spaces:** Exploiting white spaces in the licensed TV spectrum poses two major challenges. First, the unlicensed systems operating in the licensed bands must not interfere with ongoing primary usages, such as TV receptions. Therefore, these systems must have a robust mechanism for detecting primary signals and identifying white spaces. Second, the prior work has shown that white spaces are often fragmented and of different sizes, and that the availability of white spaces is temporally dynamic and depends on the geographic location of the radio [85, 98]. Consequently, a key challenge in designing the first-part solution is how to dynamically allocate white spaces to different cognitive radio nodes in a network. The spectrum allocation algorithm not only determines network throughput but the overall spectrum utilization.

**Improving WLAN:** There are two main challenges facing this part of the solution. First, the fixed channel concept has been widely adopted in WLANs.

The recent work has extensively studied the problem of channel assignment by modeling it as variances of graph coloring. However, this approach does not apply to the new spectrum allocation problem when the limit of channel bandwidth is eliminated. Therefore, a new scheme that considers the spatial and temporal disparity of traffic distribution when assigning the spectrum to APs is required. Second, as WLANs become increasingly popular, they have to scale to different client device populations, increasing physical-layer speed, and protocol overhead. Specifically, APs deployed in different scenarios need to serve different user sizes and accommodate considerable variations in the number of active clients. The user population can range from a few in a home, to tens in a community or enterprise networks [18, 23, 25, 51, 95]. Moreover, wireless clients may use a wide range of wireless devices with different communication capabilities. However, the issue of designing scalable protocols for accessing the spectrum has not been adequately addressed [101].

To addressing these challenges, we propose the novel concept of dynamic spectrum allocation, which divides available spectrum into non-overlapping frequency segments of variable bandwidth depending on the number of potentially interfering transmissions and the distribution of traffic load in a local environment. The proposed solution is based on this concept, and thus it can adaptively tune radio systems to operate in white spaces and dynamically adjust radio parameters to reflect the distribution of traffic load in a WLAN. Since existing radio systems do not support such a dynamic solution, we exploit a highly flexible radio system called cognitive radio, which makes it possible to scan the spectrum and reconfigure key operating parameters, such as center-frequency, bandwidth and power, with a very low time overhead. By sensing and adapting to its environment, cognitive radios are able to fill voids in the wireless spectrum and

dramatically increase spectral efficiency. Although the gains to be made by the combination of cognitive radios and the dynamic spectrum allocation seem intuitive, the wireless community lacks research on the wireless protocol stack to enable this coupling. To bridge this gap, this dissertation provides a complete solution comprised of radio design, scalable spectrum access mechanism, and dynamic spectrum allocation algorithm.

This dissertation is based upon the hypothesis that the proposed concept of dynamic spectrum allocation improves spectrum efficiency. In the nutshell, our proposed dynamic spectrum allocation stipulates the following rules: (1) cognitive radios access only the segments of the licensed spectrum unoccupied by primary users, which ensures non-disruptive spectrum usages for primary operators. (2) The amount of spectrum allocated to a cognitive radio node is decided dynamically as a function of total bandwidth of available spectrum and traffic loads at the node and at the neighboring nodes. More specifically, the available spectrum is divided into non-overlapping frequency segments with larger bandwidth if there are few users in a network. When there are more competing users, the smaller segments of spectrum are allocated to each user to reduce co-channel interferences and signaling overheads at each spectrum segment. This rule makes certain that the available spectrum is extensively exploited regardless of the number of contending users in a network. Consequently, the dynamic spectrum allocation largely reduces spectrum wastage and co-channel interferences as compared to the fixed allocation schemes.

For enabling this paradigm of dynamic spectrum allocation, we first develop a prototype platform of the cognitive radio. The prototype design consists of a scanner and a reconfigurable radio. The scanner periodically scans the predefined band and searches for idle frequencies. While not scanning, the scanner functions as a receiver listening for control packets on a fixed control channel. The

reconfigurable radio can be tuned to operate on certain center-frequency, and it supports several bandwidth options and power levels. We then propose a networking system of cognitive radios, called KNOWS (Kognitiv Networking Over White Spaces), as the first part of solution. The KNOWS system can reliably detect white spaces in the TV bands using collaborative sensing, and efficiently use these white spaces by enabling the dynamic allocation. For the second part of the solution, we propose and evaluate a radically new channelization structure for WLANs that implements the proposed dynamic spectrum allocation. To enable this new channelization, we adopt a centralized controller that dynamically allocates variable size channel-widths and center-frequencies to every AP. The width of an AP's channel is determined as a function of the traffic demand at the AP and at the interfering APs in its vicinity. To support this dynamic channelization, we proposed a token-coordinated random access MAC (TMAC) to address scalability issues inherent in the existing IEEE 802.11 MAC. Through extensive simulation and analysis studies, we demonstrate that these two parts of solution significantly improve spectrum efficiency in both licensed and unlicensed frequency bands. We believe that the solution holds a great promise to tackle the spectrum shortage problem and support new wireless technologies.

## 1.1 Key Contributions

To summarize, I make five primary contributions in this dissertation:

- I propose a complete cognitive radio system called KNOWS, which enables dynamic access and efficient sharing of white spaces by adaptively allocating the spectrum among contending users. The system is a fully distributed design; and thus it does not require any centralized coordinator. This feature makes the proposed design a great candidate for building WLANs and

wireless mesh networks in the TV bands with low deployment cost. The proposed system is a synergistic design of a cognitive radio platform, a new spectrum-aware MAC protocol, and a dynamic spectrum allocation algorithm [98]. The radio platform captures main functionalities of cognitive radios including sensing capability and reconfigurability. The new MAC protocol, called Cognitive MAC (CMAC), is based on a control channel, and it incorporates "virtual sensing" to arbitrate access to the dynamic and fragmented white spaces. Specifically, I enhance the RTS (request-to-send)/CTS (clear-to-send) mechanism in IEEE 802.11, which reserves airtime on a given channel, to reserve a chuck of spectrum for a period of time using the control channel. CMAC enables all networked nodes to main up-to-date information about spectrum allocations in their neighborhood. Based on the collected information, the spectrum allocation algorithm, residing in each node, makes a real-time, adaptive and local decision on which segment of spectrum to use and for how long.

- To study the problem of dynamic spectrum allocation, we develop a theoretical framework for this problem in whitespace cognitive radio networks. The framework captures the essential features of cognitive radios such as frequency agility and adaptive bandwidth. And then I introduce a concept of a *time-spectrum block*, which represents the time for which a cognitive radio uses a portion of the spectrum. I use this concept to define the spectrum allocation problem as the packing of time-spectrum blocks in a two dimensional time-frequency space, such that the demands of all nodes are satisfied best possible. The theoretical model based on the concept of time-spectrum block profoundly changes the conventional analysis framework of using variants of graph-coloring problem for maximizing spectrum utilization. We prove NP-hardness of the problem and present a approx-

imation algorithm that assumes full knowledge of all user demands and whose performance is within a small constant factor of the optimum, regardless of the network's topology. More importantly, I propose b-SMART (distributed spectrum allocation over white spaces), the first-known practical, distributed protocol to solve the spectrum allocation problem in real cognitive networks. b-SMART enables each node to dynamically decide on a time-spectrum block based on local information only [99].

- In improving the WLAN design, I revisit channelization, which is a fundamental, yet largely unexplored, aspect in the design of WLANs. I show that among other parameters, such as transmission power and data rate, WLAN designers should consider the channel-width as a configurable parameter in the design of efficient WLANs. I expose and quantify the vast potential increase in both capacity and fairness that results from abandoning today's fixed channelization concept, and I propose a dynamic channelization structure that is capable of tapping this potential. Based on a formal definition of the resulting optimization problem, we devise and evaluate novel algorithms that dynamically and flexibly allocate channels of variable width to different APs. These algorithms are computationally efficient and directly applicable in practice as they operate under the constraints implied by today's available cognitive radio platforms.

- As a component of the second part of solution, I design TMAC to effectively regulate the medium access in each wireless cell [101]. The proposed design scales to various population sizes and a wide range of high physical-layer rates. TMAC employs centralized, coarse-grained channel coordination at the higher tier, and distributed, fine-grained random access at the lower tier. The higher tier organizes stations into multiple token groups

and permits only the stations in one group to contend for the channel at a time. This token mechanism effectively controls the maximum intensity of channel contention and gracefully scales to diverse population sizes. At the lower tier, the distributed random access procedure is exploited within an adaptive channel sharing model, which largely reduces protocol overhead and aggressively exploiting rate diversity among stations. TMAC serves as the MAC-layer support for assigning different channel-widths to APs.

- I extensively analyze and evaluate each part of the solution to demonstrate its effectiveness. The simulations are performed in the network simulator QualNet [81]. I extend the simulator to closely capture the capabilities and constraints of the existing hardware. For the first part of solution, I implement the complete KNOWS design including CMAC and b-SMART in QualNet. The simulation results demonstrate that in most common scenarios, KNOWS increases the system throughput by more than 200% when compared to an IEEE 802.11 based system [98]. We also analyze the performance of KNOWS in the proposed theoretical framework. Our analysis closely matches the simulation results. Finally, we analyze and simulate the second part solution. The results are obtained using extensive simulations that are triggered by real data traces from large enterprise/campus WLAN deployments, as well as a network with user mobility. Through detailed studies, I demonstrate that all our algorithms for allocating variable channel-widths significantly improve the capacity and per-client fairness of IEEE 802.11 networks [100], and that the network throughput achieved by TMAC scales to various client sizes and physical-layer rates.

## 1.2 Organization of Thesis

This dissertation is organized as follows:

Chapter 2 consequentially elaborates (1) recent trends of using white spaces and improving WLANs, (2) limitations of existing systems in exploiting white spaces and enhancing WLANs using concrete examples and simulation studies, (3) a cognitive radio platform for supporting spectrum sensing and radio reconfiguration, and (4) the concept of dynamic spectrum allocation and its supporting mechanisms.

Chapter 3 comprehensively describes the KNOWS system that enables opportunistic access and efficient sharing of white spaces by adaptively allocating the spectrum among competing nodes. This chapter first outlines the system architecture; and then a new cognitive MAC design and a spectrum allocation algorithm that implements the dynamic allocation concept are presented. Chapter 3 also includes a theocratical formulation of the spectrum allocation problem in white spaces, which captures main features of white spaces and capabilities of cognitive radios. Finally, the chapter demonstrates the performance of the KNOWS system through analysis of the proposed spectrum allocation algorithm within the theocratical framework and extensive simulations.

Chapter 4 presents a novel channelization structure for WLANs based upon the dynamic spectrum allocation and a scalable MAC design for supporting this new channelization. This chapter first formally formulates the problem of dynamic allocating variable-width channels among potentially interfering access points within a centralized environment. And then three channel allocation algorithms running in the central controller are presented. Following that, we show the performance of these algorithms through simulation studies triggered by real traces of WLANs, and then discuss practical issues and related work. Furthermore, we present the scalable MAC design and its performance measured by

network throughput and fairness. Finally, we discuss alternative designs and related work of the proposed MAC scheme.

Chapter 5 concludes the dissertation by summarizing our main contributions. This chapter also sheds light on other directions for improving spectrum efficiency, such as market-based approaches via spectrum leasing and secondary market.

## 1.3 Dissertation Insight

If we would condense the dissertation into one paragraph, the paragraph is as follows: The key insight of this dissertation is the proposed concept of dynamic spectrum allocation, which divides available spectrum into the appropriate number of non-overlapping frequency segments to accommodate potentially interfering transmissions, and assigns each segment with variable bandwidth depending on traffic distribution in a local environment. The allocated segments are dynamically adjusted to handle variations of user demands. In this way, the proposed allocation scheme enables opportunistic access and efficient sharing of available spectrum. More specifically, the dynamic spectrum allocation maximizes spectrum efficiency by allocating frequency segments with larger bandwidth when few users present, and by creating smaller width channels to maximize parallel transmissions and reduce interferences and signaling overhead, when the user population is large. The dynamic spectrum allocation also improves fairness across a network since it balances spectrum assignments by allocating heavily loaded nodes with more spectrum. Through building wireless systems based upon this concept, we demonstrate significant improvements in spectrum utilization and efficiency achieved by these systems, and thus confirm the effectiveness of the proposed concept of dynamic spectrum allocation.

# Chapter 2

# Foundation: Cognitive Radios, Dynamic Spectrum Allocation

*A hypothesis is ...*
*a mere trial idea,*
*a tentative suggestion concerning the nature of things.*
*Until it has been tested,*
*it is should not be confused with a law... .*
*Plausibility is not a substitute for evidence,*
*however great may be the emotional wish to believe.*
*— E. Bright Wilson, Jr.*

Cognitive radios, a term first coined by Mitola [47], plays an essential role in translating the concept of dynamic spectrum allocation into true spectral efficiency gains. These radios can sense the spectrum to identify primary usages and white spaces. The operating parameters of cognitive radios, such as center-frequency, bandwidth, and power, can be reconfigured with a very low time overhead (less than 50 microseconds) [98]. Cognitive radios, therefore, are flexible enough to fill voids in the wireless spectrum and dramatically increase spectral

efficiency. In this chapter, we first outline two recent trends in Section 2.1. These trends motivated us to exploit cognitive radios for enabling the proposed dynamic spectrum allocation in filling white spaces in the licensed spectrum and improving the efficiency of WLANs in the unlicensed bands. In Section 2.2, we present our cognitive radio platform and its performance, and then we discuss other existing designs and implementations of cognitive radios. Finally, we explain our hypothesis about the proposed dynamic spectrum allocation scheme and its supporting mechanisms in a great detail in Section 2.3.

## 2.1 Two Recent Trends

### 2.1.1 Trend 1: White Spaces

Recent measurement results have consistently shown that the licensed spectrum is significantly under-utilized [33,85]. This indicates the abundance of white spaces, unoccupied frequencies in the licensed spectrum. At the mean time, the explosive success of unlicensed operations and the many advancements in technology that resulted from it, led regulatory bodies (e.g., the FCC) to consider opening white spaces for unlicensed use. On May 12, 2004, the FCC issued a Notice of Proposed Rule Making (NPRM) proposing unlicensed use of unused TV channels 2-to-51 (Docket 04-186). When the Digital TV (DTV) transition ends in early 2009, most of the nations 210 TV markets will have between 10 and 40 unassigned channels reserved for broadcasting, but not in use. The FCC proposal would allow a new generation of wireless broadband devices to utilize the vacant TV channels in each local market for Wi-Fi and other unlicensed technologies.

One import part of the NPRM is to regulate unlicensed usages to avoid causing interferences to the primary users in the licensed bands. For the purpose of developing interference protection criteria, the NRPM proposes to classify the

unlicensed broadband devices to be used in the TV bands into these two general functional categories. The first category will consist of lower power "personal/portable" unlicensed devices, such as Wi-Fi like cards in laptop computers or wireless in-home LANs. The second category will consist of higher power "fixed/access" unlicensed devices that are generally operated from a fixed location and may be used to provide a commercial service such as wireless broadband internet access. The fixed access devices are required to incorporate a method for determining geographic location with a minimum accuracy of 10 meters. To meet this requirement, for example, the device could incorporate a GPS receiver to determine its geographic coordinates. Using this location information, local broadcast station data and the protection requirements described, channel availability for the unlicensed device can be determined. The maximum conducted output power for fixed devices is 1 watt peak, and the maximum conducted output power for portable devices is 100 milliwatts peak. The NRPM further requires that portable devices have a permanently attached integral antenna with a maximum permissible gain of 6 dBi.

Although the strict regulating rules are in place, we believe that these power and antenna provisions provide sufficient communications capabilities to allow personal/portable broadband devices to serve a wide range of broadband applications, such as home networks, LANs and broadband connectivity. Especially the TV bands are in the lower frequencies, which deliver longer transmission ranges, and better penetration prosperities as compared to the unlicensed bands in the high frequencies. These desirable features make the TV spectrum ideal for wireless data communications.

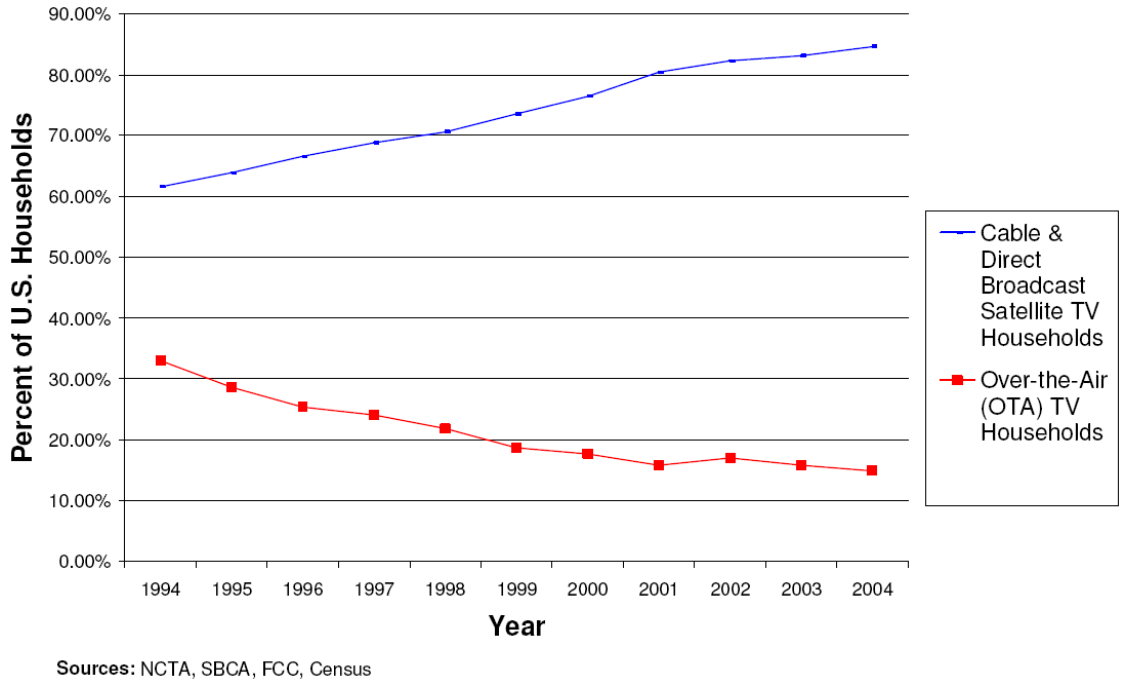


Figure 2.1: The decline of over-the-air television

#### 2.1.1.1 Abundance of White Spaces in the TV bands

The white spaces in the TV spectrum are abundant. How much vacant TV band spectrum is available around the country? There are 210 local TV markets in the United States. Each is currently allocated with channels among the 67 channels (channels 2-to-69, excluding channel 37, which is reserved for radio astronomy and medical telemetry). The FCC's current TV allocation plan mandates that after the DTV transition, channels 52-to-69 will be cleared of TV signals in all 210 local TV markets in the United States. Four of these channels are being reallocated for public safety agencies, while ten others will be auctioned for exclusive, licensed use by commercial wireless firms. In February 2009, at the end of the DTV transition, broadcasters must give back one of their two channels. Consequently, even after channels 52-to-69 are returned, substantial guard band spectrum will remain, especially in small TV markets, on the 49 channels from channels 2-to-51 [52].

Notably, the demands for TV broadcasting services are decreasing every year. Figure 2.1 depicts the decline of terrestrial over-the-air TV and the rise of cable and satellite TV. From 1970 to 2005, the percentage of US television households relying exclusively on over-the-air reception for their TV has declined from essentially 100% to less than 13% [21], with a drop of about 14 percentage points coming in the last decade alone [32]. This drop is remarkable, since it occurred despite huge government subsidies to preserve over-the-air TV and despite the fact that an additional fee is required to view identical local broadcast TV programming over cable or satellite systems. So far, the figures for digital over-the-air TV are even more dismal. As of 2004, 40.4% of Americans had access to digital TV but only 2.7% of those relied on broadcast TV. The rest relied on cable DTV (50.7%) and satellite DTV (46.6%) [73]. The decreasing demand for TV broadcasting services further gives rise to exploiting the white spaces for data communications.

#### 2.1.1.2 Features of White Spaces in the TV bands

The white spaces in the TV spectrum have several features, which we summarize as follows:

- Low-frequency spectrum is better suited for mobility because its waves are longer and can thus better pass through objects such as walls, foliage and weather [28]. All terrestrial mobile telephone services, for example, are located below 3 GHz (the lowest 1% of the radio spectrum). If cell phone service went out every time someone passed a tree or building, its utility would be minimal. Similarly, Wi-Fi services would be much less valuable if it couldn't pass through walls, furniture, people and other common household obstructions. And the cost and quality of wireless broadband deployments would improve dramatically if networks and devices could operate below 1 GHz, where TV operates today.

- Another major physical advantage of low-frequency spectrum is that it requires less energy than high-frequency spectrum to cover the same distance. The large waves that characterize low-frequency spectrum lose less energy when they pass through objects. As a result, they can cover greater distances with the same power. This, in turn, means that battery-powered devices can be less expensive, longer-lived, smaller and lighter. In the emerging era of ubiquitous, portable wireless devices, this can be a great advantage.

- The availability of white spaces changes over time and depends on the geographic location of the radio [85, 98]. In U.S., the TV channel allocations are varied in different locations, and the TV broadcasting stations are deployed based on user populations and distributions. Therefore, the number of vacant TV channels and the frequencies of vacant channels are different across different regions. Typically there are more vacant TV channels in rural areas than urban ones. Moreover, the vacant TV channels changes over time depending on the diurnal schedules of TV broadcasters.

- The white spaces are often fragmented by primary users. As shown in Figure 2.2, primary TV signals divide the spectrum into noncontiguous segments, each of which may have a different size. The bandwidth of a contiguous segment depends on the allocation of TV channels in that region.

#### 2.1.1.3 Cognitive Radio for White Spaces

On one hand, the white spaces are abundant and desirable resources for data communications. On the other hand, the dynamic and fragmented feature makes the white spaces radically different from the unlicensed spectrum, which is the segment of static and contiguous spectrum. Consequently, the wireless sys-

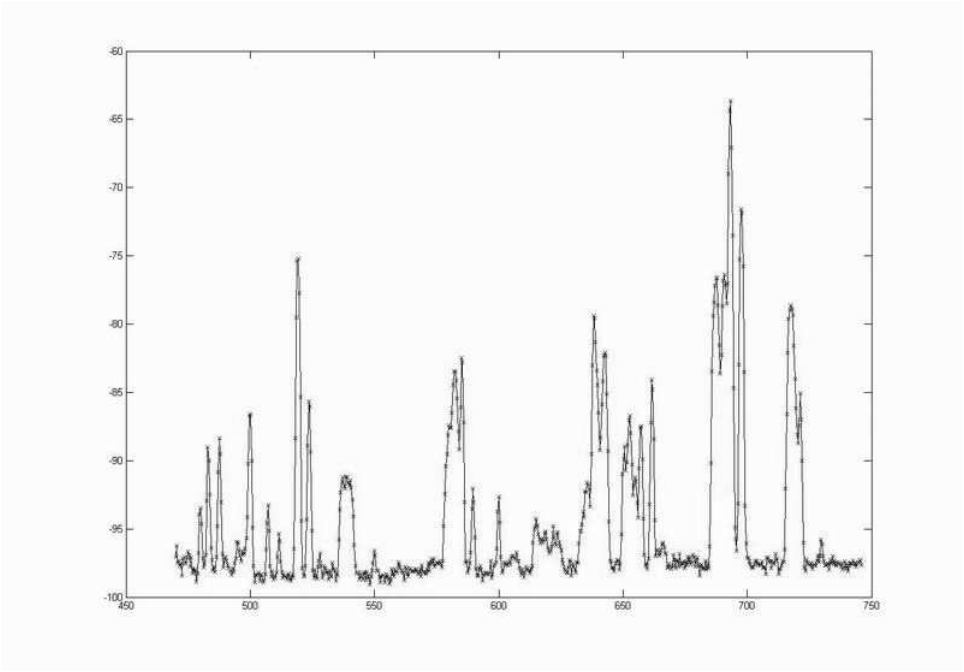


Figure 2.2: The signal strength measured in the TV spectrum

tem and the spectrum allocation scheme designed for the unlicensed spectrum do *not* directly apply to the white spaces. In specific, since the unlicensed spectrum is static and has no fragmentation, it is divided into several fixed channels of equal bandwidth. For example, the 2.4 GHz ISM band has 3 non-overlapping channels, and each channel is 22 MHz wide. The wireless device can choose a particular channel to operate depending on the connectivity requirement in the network. The competing wireless devices on the same channel are regulated using IEEE 802.11 MAC. The center-frequency and the bandwidth of any channel are pre-assigned and fixed. However, the fixed spectrum allocation scheme is not suitable for the white spaces. First, the white spaces are dynamic. It is required to first detect those white spaces before any allocation. Second, the channel is not well-defined in the white spaces. One could argue that the white spaces can adopt the channelization defined by TV broadcasters. However, the fixed allocations leads to the inefficiency usage of white spaces. The reasons are as follows. If the channel bandwidth is defined to be smaller, for example, each TV channel is

6 MHz in the U.S., it is desirable to combine multiple channels together to deliver high data rates if there are few contending users. While more competing transmissions take place, dynamically reducing bandwidth of each channel creates more channels, which achieves high efficiency by improving parallelism and reducing interferences. If the channel bandwidth is defined to be larger, such as 22 MHz, any primary operation in between renders the entire 22 MHz channel useless. Therefore, the center-frequency and the bandwidth of a channel should be defined based on the instantaneous information of the radio environment, such as primary signals, the number of competing transmissions, and traffic demands of users.

Cognitive radio is a promising technology enabling such a real-time allocation of spectrum. Equipped with a spectrum scanner, the cognitive radio can periodically scan the spectrum to identify the white spaces, and the radio parameters can be reconfigured to operate in the white spaces. Therefore, it is essential to design a reconfigurable radio platform with the scanning capability, and an algorithm to drive the radio by deciding and reconfiguring the radio parameters. We will elaborate our radio platform design in Section 2.2, and the concept of dynamic spectrum allocation in Section 2.3

### 2.1.2 Trend 2: Disparity in WLANs

The recent measurement results of usage patterns in real-world WLANs, including conference settings, university and corporate campuses, have revealed that the spatial and temporal disparity in client distributions [18, 23, 25, 95] in large-scale WLANs is prevailing. For example, a recent study of IBM's WLAN consisting of 177 APs [18] shows that 40% of the APs never have more than 10 active clients, while a few APs in auditoriums and cafeterias have 30 simultaneously associated users. The study also shows that the set of heavily loaded APs

changes over time, but the current practice of assigning fixed-width channels and the MAC design in IEEE 802.11 networks does not take into account such spatial and temporal disparity in client distributions.

We identify two main problems caused by the usage disparity. First, under the assumption of uniform traffic distribution across the network, IEEE 802.11 divides the spectrum into a fixed number of channels with equal channel width, which is 22 MHz wide in IEEE 802.11b/g and 20 MHz wide in IEEE 802.11a. However, in dynamic conditions, the adherence to fixed-width channels can be problematic and suboptimal. When the number of APs is fewer than the number of available channels, the spectrum is not fully utilized since each AP uses only one channel. On the other hand, if the number of APs is large, two or more neighboring APs are inevitably assigned the same channel, which limits parallelism and creates interferences [14].

Second, WLANs are being deployed in much more diversified environments. In some of these scenarios, each access point has to support a much larger user population and be able to accommodate considerable variations in the number of active stations. The wireless protocols should not constraint the number of potential users handled by a single AP. However, the performance of current MAC proposals [16, 55, 75, 88, 102] does not scale as user population expands. Specifically, at user population of 300, the 802.11 MAC not only results in 57% degradation in aggregate throughput but also leads to starvation for most stations, as shown in our simulations. Moreover, WLANs (e.g., IEEE 802.11n [6]) promise to deliver much higher data rates in the order of 100s of Mbps [7], through advanced antennas, enhanced modulation and transmission techniques. This requires MAC-layer solutions to develop in pace with high-capacity physical layers. However, the widely adopted IEEE 802.11 MAC [75], using distributed coordination function (DCF), does not scale to the increasing

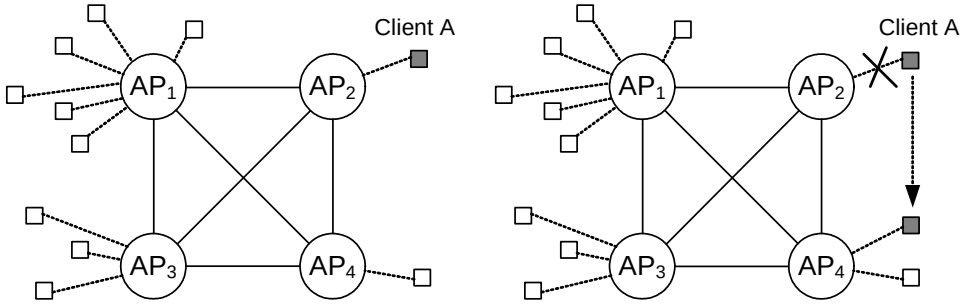


Figure 2.3: A network with four mutually interfering APs. With fixed channel bandwidths, both throughput and fairness is suboptimal.

physical-layer rates. According to our analysis and simulations, DCF MAC delivers as low as 30Mbps throughput at the MAC layer with the bit-rate of 216Mbps, utilizing merely 14% of channel capacity.

To illustrate the problems caused by the fixed channels and the 802.11 MAC in WLAN, we first give an example to show the constrains of fixed channels in Section 2.1.2.1. Then through analysis and simulations, we demonstrate the limiting factors of IEEE 802.11 MAC in Section 2.1.2.2. Finally we propose using cognitive radios to improve the efficiency of WLANs in Section 2.1.2.3.

#### 2.1.2.1 Fixed Channels v.s. Dynamic Channels: An Example

Figure 2.3 illustrates the scenario with four APs all within mutual interference distance of one another. In case 1, $AP_1$ has 6 clients, $AP_3$ has 3 clients, while the remaining two APs have one client each. In case 2, client A moves away from $AP_2$ and associates to $AP_4$. We compare the performance of using the fixed channels (F) with dynamic channels (D). In the fixed channel case, the spectrum is divided into 4 channels of 20 MHz each. In the dynamic channel case, channels may be 10, 20, or 40 MHz. Table 2.1 lists the bandwidth received *per client* at each AP. Also included is the total bandwidth used (B), and Jain's fairness index (FI). The index is calculated using $(\sum c_i)^2/n\sum c_i^2$, where $c_i$ is the bandwidth obtained by client $i$, and $n$ is the total number of clients.

Table 2.1: Bandwidth received by each client (normalized by 20MHz)

| Scenario | $AP_1$ | $AP_2$ | $AP_3$ | $AP_4$ | B | FI |
|---|---|---|---|---|---|---|
| Case 1:F | 1/6 | 1 | 1/3 | 1 | 4 | 0.58 |
| Case 1:D | 2/6 | 1/2 | 1/3 | 1/2 | 4 | **0.97** |
| Case 2:F | 1/6 | X | 1/3 | 1/2 | 3 | 0.82 |
| Case 2:D | 2/6 | X | 1/3 | 1/2 | **4** | **0.97** |

In case 1, the fixed-width channelization leads to severe un-fairness among different clients. The client in the crowded location ($AP_1$) receives 1/6 of bandwidth compared to the client associated with $AP_2$ and $AP_4$. In contrast, with an allocation of 40 MHz to $AP_1$, 20 MHz to $AP_2$ and 10 MHz to the remaining APs, fairness improves significantly to 0.97.

Flexible and dynamic channelization is not only important for fairness, but also for system capacity. For instance, in case 2 if client A moves from $AP_2$ to $AP_4$, an adaptive approach can reallocate the 10 MHz spectrum formerly used by $AP_2$ to $AP_4$ (thus giving $AP_4$ a total of 20 MHz).

Our study of real-world traces shows that in a large corporate and university wireless networks fairness and capacity problems illustrated in Figure 2.3 occur frequently. Therefore, to support the growth of WLAN and deliver uniform services to wireless users, it is critical to re-examine the fixed channelization scheme to handle the temporal and spatial disparity in WLAN usages.

#### 2.1.2.2 Scalability of IEEE 802.11 MAC

We consider the scalability issues in wireless MAC protocols along the following three dimensions.

**Network Capacity:** Advances in physical-layer technologies have greatly improved the link capacity in wireless LANs. The initial 1∼11Mbps data rates specified in 802.11b standard [75] have been elevated to 54Mbps in 802.11a/g [42],

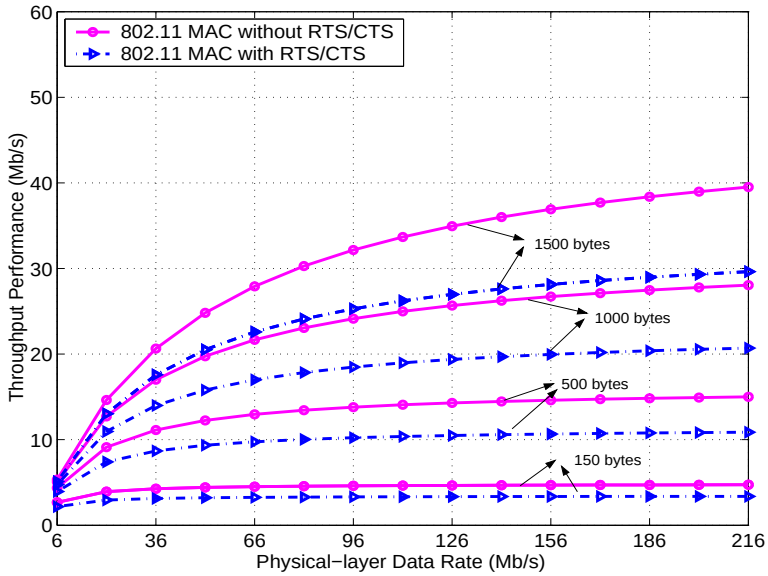


Figure 2.4: Throughput achieved by DCF MAC at different physical-layer data rates

and to 100s of Mbps in 802.11n [6]. Therefore, MAC-layer throughput must scale up accordingly. Furthermore, MAC designs need to exploit the multi-rate capability offered by the physical layer for leveraging channel dynamics and multi-user diversity.

**User population:** Another important consideration is to provide efficient solution regardless of the number of contending stations. The user population may range from a few in an office, to tens or hundreds in a classroom or a conference room, and thousands in public places like Disney Theme Parks [67]. As the number of active users grows, MAC designs should control contentions and collisions over the shared wireless channel and deliver stable performance.

**Protocol overhead scalability:** The third aspect in efficient wireless MAC design is to minimize the protocol overhead as the population size and the physical-layer capacity increase. Specifically, the fraction of channel time consumed by signaling messages per packet, due to backoff, inter-frame spacings, and handshakes, must remain relatively small.

In general, both CSMA/CA [75] and polling based MAC solutions have

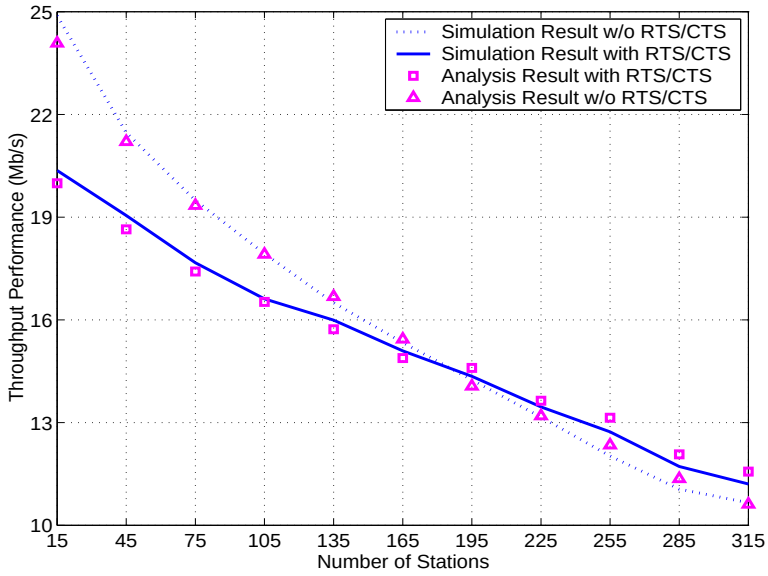


Figure 2.5: Throughput achieved by DCF MAC at different user populations

scalability limitations in these three aspects.

### CSMA/CA Based MAC

Our analysis and simulations show that IEEE 802.11 Distributed Coordination Function (DCF) MAC, based on CSMA/CA mechanism, does not scale to high physical-layer capacity or various user populations. In CSMA/CA, upon a packet transmission, a station uses carrier sensing, waits until idle channel status, and then defers for a DIFS time interval. The station then backs off for a value randomly chosen between zero and Contention Window (CW). Once the back-off timer expires, an optional RTS/CTS handshake is initiated between the two stations, followed by a data packet and a ACK transmission. The transmissions of RTS, CTS, data, and ACK are are separated by SIFS time interval. Upon each unsuccessful transmission, CW is doubled up to a maximum value. In the infrastructure mode, when the packet size is larger than the RTS threshold, RTS/CTS is recommended to be turned on to reduce the duration of collisions [75].

The throughput of a single station in the channel without RTS/CTS is spec-

ified by [30]:

$$S = \frac{E[P]}{T_{DIFS} + E[B] + E[P]/R + T_{SIFS} + T_{ACK} + 2\delta} \tag{2.1}$$

where $E[P]$ is the average size of packet payload, E[B] denotes the average back-off window expressed as $E[B] = (CW_{min} - 1)/2$, and $R, \delta$ are the current transmission rate and propagation delay, respectively.

We plot the theoretical throughput attained by DCF MAC with different packet sizes in Figure 2.5.a. Table 4.3 lists the values of DIFS, SIFS, ACK, MAC header, physical-layer preamble and header according to the specifications in [7, 42]. Note that DCF MAC delivers at most 40Mbps throughput without RTS/CTS at 216Mbps, which further degrades to 30Mbps when the RTS/CTS option is on. Such unscalable performance is due to two factors. First, as the link capacity increases, the signaling overhead ratio grows disproportionately since the time of transmitting data packets reduces considerably. Second, the current MAC adopts a static channel sharing model that only considers transmission demands of stations. The channel is monopolized by low-rate stations. Hence the network throughout is largely reduced. Figure 2.5.b shows results from both analysis [1] and simulation experiments as the number of users increases from 15 to 315. The users transmit UDP payloads at 54Mbps. The network throughput obtained with DCF reduces by approximately 50% as the user population reaches 300. The significant throughput degradation is mainly caused by dramatically intensified collisions and increasingly enlarged contention window (CW).

[1] We employ analytical model proposed in [34] to compute throughput, which matches the simulation results.

### Polling-based MAC

Polling-based MAC schemes [40, 43, 88] generally do not possess capacity and protocol overhead scalability due to the excessive polling overhead. To illustrate the percentage of overhead, we analyze the polling mode (PCF) in 802.11b. In PCF, AP sends the polling packet to initiate the data transmission from wireless stations. A station can only transmit after receiving the polling packet. Idle stations response the polling message with NULL frame, which is a data frame without any payload.

We use the method proposed in [80] to analyze the polling mode (PCF) in 802.11b, which specifies the percentage of protocol overhead as

$$P_o = 1 - \frac{E[P]}{R(p \cdot T_w/(1-p) + T_s)},$$

and the average delay as $D_{avg} = N(T_s - (T_s - T_w)p)$, where $p$ represents the possibility of having no pending data upon receiving the polling. $N$ is the number of stations on the polling list. $T_w = T_{poll} + SIFS + T_{null} + SIFS$ and $T_s = T_{poll} + SIFS + E[P]/R + SIFS + ACK + SIFS$, where $T_{poll}$ and $T_{null}$ are the time for sending polling and null frame, respectively. Table 2.2 lists the protocol overhead as the fraction of idle stations increases. We compute the results using the values of DIFS, SIFS, ACK, MAC header, physical-layer preamble and header listed in Table 4.3, and the packet size used in this study is 1.2Kb.

Note that the overhead ratio reaches 52.1% even when all stations are active at the physical-layer rate of 54Mbps, and continue to grow considerably as more idle stations present. Furthermore, as the link capacity increases to 216Mbps, over 80% of channel time is spent on signaling messages.

Table 2.2: Polling overhead vs. percentage of idle stations

| | 0 | 15% | 30% | 45% | 60% |
|---|---|---|---|---|---|
| 54Mbps | 52.1% | 55.2% | 59.1% | 64% | 70.3% |
| 216Mbps | 81.6% | 83.2% | 85.5% | 87.3% | 90.4% |

#### 2.1.2.3 Cognitive Radios for WLAN

These examples motivate the need for a dynamic spectrum allocation and a scalable MAC design in IEEE 802.11 networks to cope with various *spatial distributions of load* as well as *temporal changes of the load due to mobility*.

However, the existing wireless systems, such as Wi-Fi, are designed with the fixed channelization in heart. Although the agility of the their physical layers is improved, for example, a wireless device can switch from one channel to the other in less than 80 $ms$ [76], these wireless systems still do not support variable bandwidths and have limited reconfigurability. To underpin the potentials of the dynamic spectrum allocation, a new radio system and a dynamic channelization structure are required. Cognitive radio, whose operating parameters can be adjusted in real-time, is a promising technology for improving WLANs. Since cognitive radio is a new concept, we need to not only set up the radio platform, but design algorithms of dynamic spectrum allocation to configure the radio parameters based on the local environment. At the same time, the enhanced MAC scheme should scale to the increase of user population and physical-layer rate with certain fairness notion, as some AP with larger bandwidth may serve a large number of clients and client devices may have different communication capabilities.

## 2.2 Cognitive Radio Platform

Through previous discussions, we show that a cognitive radio is a promising technology to improve spectrum utilization by dynamically accessing and allocating spectrum. Since there is no standardized design for such a new radio system, we develop our cognitive radio platform based on the requirements of our proposed dynamic spectrum allocation. In brief, the radio platform can operate in the TV spectrum and the 2.4 GHz ISM band. A scanner is incorporated in the platform to detect primary signals in the TV spectrum, such as DTV operations and microphones; and the upper layer, i.e., MAC, can directly control radio operating parameters, such as center-frequency, bandwidth and power. The platform is designed to provide a great flexibility in order to support dynamic spectrum allocation schemes.

In the section, we sequentially present the physical-layer design, the implementation and the performance of our cognitive radio platform.

### 2.2.1 Cognitive Radio Design

Figure 2.6 shows the hardware platform we have built as basis for supporting dynamical access and allocation of available spectrum. The platform consists of four main function blocks, namely the reconfigurable radio, the scanner radio, the GPS receiver, and the x86 embedded processor.

The reconfigurable radio has a set of operational parameters that can be adjusted with low time overhead. The current implementation of the reconfigurable radio uses a commodity IEEE 802.11g card to generate the OFDM signals at 2.4GHz. We use a wide band frequency synthesizer to convert the received signals to the specified frequency. To control the reconfigurable radio, the interface to the MAC layer is a list of register values that specifies the operating frequency,

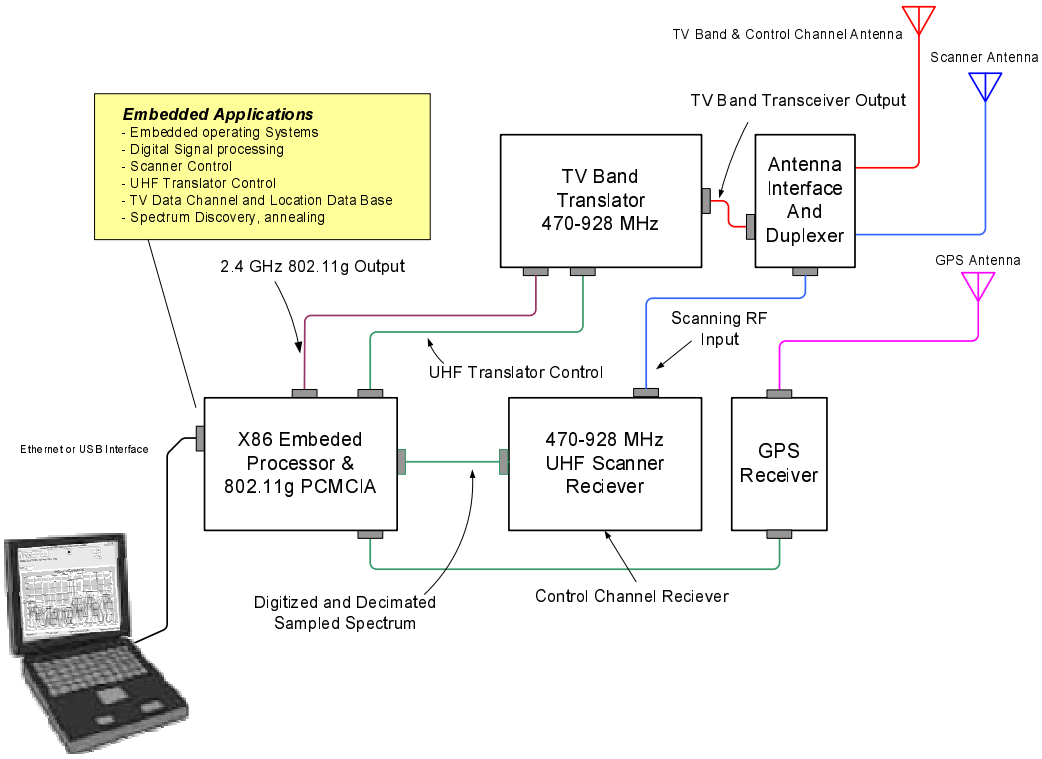


Figure 2.6: Cognitive radio development platform

bandwidth, and transmission power level. The operating frequency can be set from 400 to 928 MHz and from 2.412 GHz to 2.484 GHz in 0.5 MHz steps. The set of bandwidth options is 5, 10, 20, and 40 MHz. The narrow bandwidth options, such as 5 MHz, are provided to use white-space spectrum in between the incumbent operators. The maximum output power is 200 mW and the power level is controllable from -8 to +23 (dBm). The threshold for packet reception in the TV band is -85 (dBm). The time overhead for adjusting the radio parameters, e.g. frequency, bandwidth, and power level, is within 100 $\mu$s in the current development board.

The scanner periodically scans the spectrum and locates the vacant pockets of spectrum without primary signals. The scanning algorithm is prototyped in the $C$ programming language with a Python-based interface and will be implemented on DSPs (digital signal processor) or FPGAs (field programmable gate

array) in the future. The scanner measures the signal power at a frequency range with a typical resolution of 3 KHz. On average, the scanner takes at most 10 ms to scan one 6 MHz TV channel. The current setting of the DTV pilot tone detection sensitivity is -115 dBm. As required by the FCC, the TV spectrum needs to be scanned once every 30 minutes, since the TV signal arrives and leaves in a very coarse time level (several hours). Therefore, for most of the time, the scanner works as a receiver operating on the control channel. The upper layer can control the scanning schedule and set the frequency range to scan by configuring the registers in the scanner.

Additionally, a GPS receiver is incorporated in the hardware board for loading location information and performing time synchronization. Based on the estimated location, the node could identify the unused spectrum in case a database with TV program information was available. This is an alternative approach suggested by the FCC for detecting incumbent users. Therefore, the GPS receiver extends the flexibility of our development platform.

The x86 embedded processor controls all radios on the platform. It takes instructions from the device driver to configure the radios, and passes packets between the host computer and the development board. The driver implements the MAC design and the spectrum allocation algorithm.

### 2.2.2 Radio Development Board and A Scanning Result

We are building a prototype based on the design of cognitive radio for conducting field experiments. We have currently implemented the radio design in the development boards, and conducted experiments to examine the functionality of the scanner in the San Diego area.

Figure 2.7 show the development boards for the scanner radio and the reconfigurable radio. The interface to the host computer is USB. Figure 2.8 shows

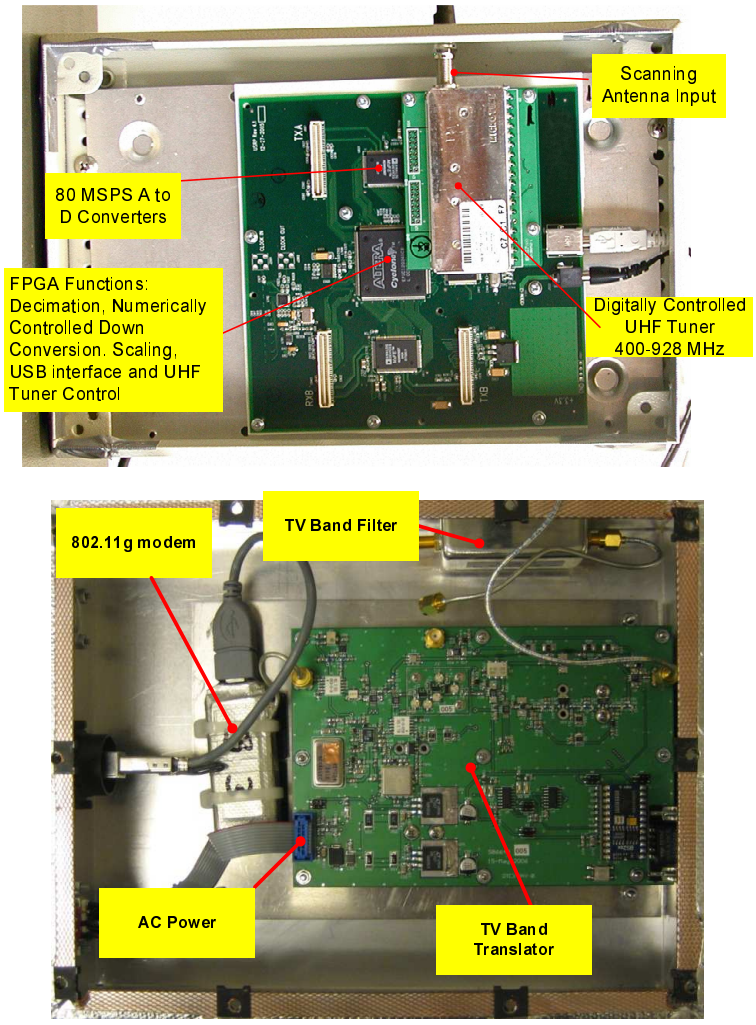


Figure 2.7: KNOWS radio system development board: scanner receiver development board (top) and reconfigurable radio development board (bottom)

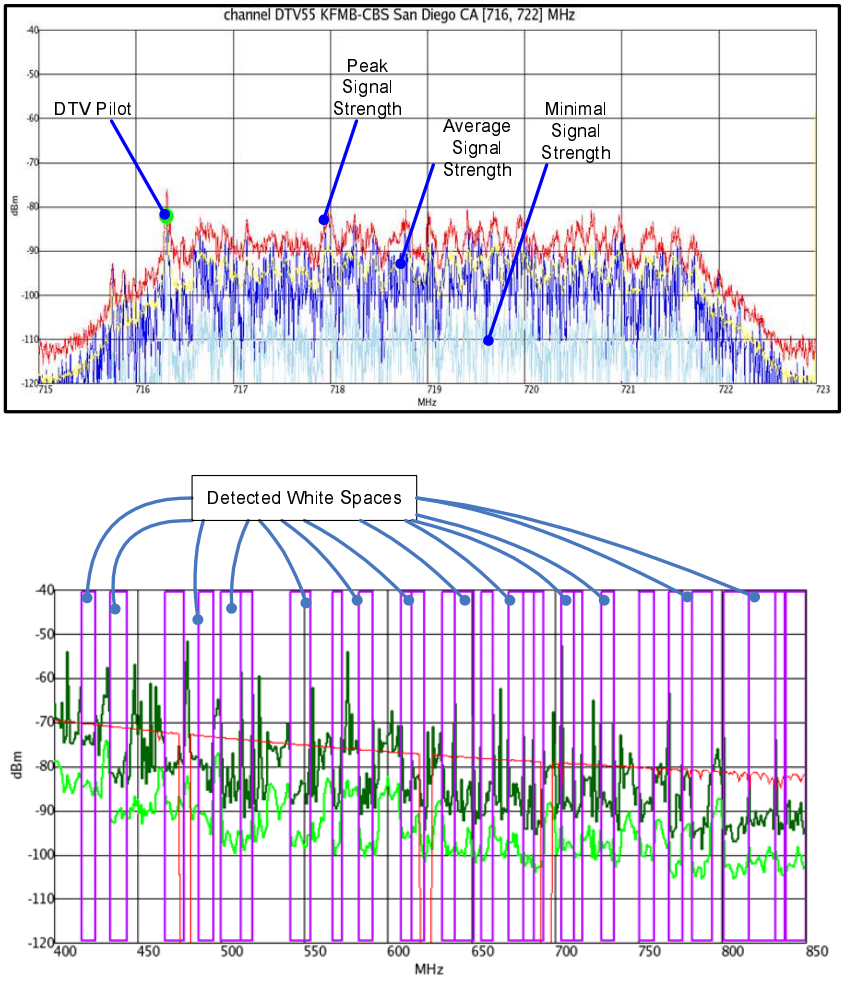


Figure 2.8: Sample scanning results in the San Diego Area: scanning the 6MHz-wide window (top) and scanning the UHF spectrum (bottom)

one sample output of the scanning operations conducted in the San Diego area. The scanner measures the signal strength of the spectrum. The top figure in Figure 2.8 shows one detected DTV pilot tone and the peak, average and minimal signal strength in the sliding 6 MHz-wide window. The bottom figure in Figure 2.8 illustrates the aggregated information for the whole TV bands using the accumulated peak and average values. We save all white spaces that are larger than 1 MHz and display white windows with at least 5 MHz bandwidth. Our measurement results confirm that the TV spectrum is fragmented especially in the metropolitan areas, while in rural areas more contiguous spectrum is available [85].

### 2.2.3 Other Implementation of Cognitive Radios

Since cognitive radio is a new concept, there is no consensus on how to implement it. Our platform is a simple and practical design to enable our proposed dynamic spectrum allocation, which we will explain in the next section.

Now we discuss another interesting implementation of cognitive radio: so-called software-defined radio. Software-defined radio system is a radio communication system that can tune to any frequency band and receive any modulation using programmable hardware controlled by software. It is the technique of getting code as close to the antenna as possible. The fundamental characteristic of software-defined radio is that software defines the transmitted waveforms and demodulates the received waveforms. The hardware of a software-defined radio typically consists of a superheterodyne RF (Radio Frequency) front end which converts RF signals from (and to) analog IF (Intermediate Frequency) signals, and analog to digital converter and digital to analog converters which are used to convert a digitized IF signal from and to analog form, respectively. This is in contrast to most radios in which the processing is done with either analog cir-

cuitry or analog circuitry combined with digital chips.

GNU Radio [36] is a free software toolkit for building software-defined radios. The existing platforms [24, 83] are built based on the GNU Radios. Since in this dissertation we focus on an efficient scheme for dynamically allocating spectrum, we develop our radio platform to support the allocating scheme. In stead of drilling down to the detail of defining radio waveforms, we build the radio platform using the existing hardware solution, and enhance the platform with scanning function and reconfigurability. Our cognitive radio platform can be easily reconfigured and controlled by the upper layer, which implements the dynamic spectrum allocation. Nevertheless, the software-defined radio also can exploit our allocation scheme to regulate spectrum access and allocations.

## 2.3 Dynamic Spectrum Allocation in Cognitive Radio Networks

In this section, we elaborate the hypothesis that **our proposed dynamic spectrum allocation improves spectrum efficiency**. Section 2.3.1 presents the background. Particularly, we illustrate the spectrum allocation schemes used in the cellular networks and the differences between allocating spectrum in the cellular networks and in the cognitive radio networks. Section 2.3.2 explains the concept of dynamic spectrum allocation we propose for cognitive radio networks. Finally, Section 2.3.3 discusses prerequisites to enable the dynamic spectrum allocation for cognitive radio networks.

### 2.3.1 Background

Assume that the cognitive radios can reliably detect the white spaces. Due to the deficiencies of adopting fixed channels, the cognitive radio network needs to solve the key problem of spectrum allocation: *Which node should use how wide a spectrum-band at what center-frequency and for how long?*. Since the MAC protocol regulate the access sequence of competing nodes, the spectrum allocation problem boils down to the problem of deciding, for each cognitive radio, the operating band defined by the center-frequency and the bandwidth, and the time duration of using the defined band. The efficiency of spectrum allocation determines both the network's throughput and the overall spectrum utilization.

Our hypothesis is that the proposed dynamic spectrum allocation improves spectrum efficiency. Dynamic spectrum allocation has been widely adopted in wireless networks. For example, in the cellular networks, many heuristic algorithms have been proposed to solve various flavors of spectrum allocation problem using simulated annealing [60], tabu search [45], evolutionary algorithms [46], neural networks [72] and graph coloring algorithms [58]. In the existing WLANs, the spectrum allocation problem within the fixed channelization structure is actually the channel assignment problem. Since the spectrum is divided into fixed channel of equal bandwidth in both the cellular networks and WLANs, the notion of colors is suitable to model the pre-defined channels.

However, in cognitive radio networks, we tackle the problem of dynamic spectrum allocation without the pre-allocated channels. The center-frequency, the bandwidth, and the life time of the defined channel are dynamic depending on the perceived local radio environment. We summarize the differences between the dynamic spectrum allocation in the cognitive radio networks and the allocation schemes in the cellular networks as follows.

- The cellular networks have the dedicated and licensed spectrum, which is

static and always available. However, the white spaces are dynamic depending on operations of primary usages in the licensed band. In such a dynamic environment, the fixed allocation schemes constraint network capacity and fairness. In particular, using fixed channels with a wider bandwidth causes many channels invalid due to the primary operations, while adopting the channels with a smaller bandwidth may leave a few channels unoccupied, which reduces spectrum utilization.

- The cellular networks are optimized to serve voice applications. The voice flow is typically treated as constant-bit-rate traffic with a very low bit-rate (50 kbps $\sim$ 150 kbps). Hence, each channel only requires a small segment of spectrum, and the channels of equal bandwidth can serve a voice flow well. In contrast, wireless data communications have much more diversities in terms of traffic loads, application types, and delay requirements. In catering to data communications, the spectrum allocation needs to maximize spectrum efficiency by adjusting frequency assignments according to application requirements and local contention intensity.

- The cellular networks adopt a centralized solution, where a base station schedules frequency or channel assignments. The base station has the full knowledge of the number of associated stations, their activities and channel qualities. The decision of how to allocate channels can be made based on the collected information. However, in the cognitive networks, we do *not* assume the existence of the central coordinator, and thus the spectrum allocation has to be decided by each cognitive radio in a distributed manner.

### 2.3.2 The Concept of Dynamic Spectrum Allocation

Our proposed dynamic spectrum allocation for cognitive radio networks works as follows. First, the spectrum allocation can only use the white spaces or the unlicensed spectrum; thus it mitigates interferences caused to primary users. Second, the amount of spectrum, i.e., the bandwidth, assigned to a pair of communicating nodes depends on the total available spectrum, the contention intensity, and the user traffic demand. Intuitively, when there are few users, each user is assigned a wide bandwidth for a higher data rate; when there are a number of competing users, the total spectrum is divided to accommodate more concurrent transmissions.

Determining the value of bandwidth is the key aspect of our dynamic spectrum allocation, and also poses an interesting challenge. It is fairly intuitive that when there are only few disjointed transmissions, each one uses a large bandwidth to avoid wasting the resource. The two transmissions are disjointed if they do not share either end point. However, when there are a number of users, it might not be straightforward to understand why the allocation divides the spectrum among all users, instead of allocating the total spectrum to only one user at a time. We provide three justifications for dividing the spectrum into segments of smaller bandwidth when the number of users is large.

1. A smaller bandwidth decreases the physical-layer data rate and increases the radio between the data transmission duration and the overall transmission time. Therefore, the overall spectrum efficiency is increased due to the reduced percentage of signaling overhead associated with every data transmission. For example, suppose there is a fixed 100 $\mu s$ [2] overhead associated with each transmission and the actual data transmission takes 200 $\mu s$ in a 1 MHz band and 100 $\mu s$ in a 2 MHz band. If we define the spectrum ef-

[2] $\mu s$ stands for microsecond.

ficiency as the fraction between the data transmission time and the overall transmission duration, then two parallel transmissions in two 1 MHz bands yields the spectrum efficiency of 67%, whereas a transmission in one 2 MHz band yields the efficiency of 50%.

2. The reduced bandwidth allows more parallel transmissions. Thus it effectively handles the fragmentation in white spaces and reduces interferences among concurrent transmissions. Due to the operation of primary users, the white spaces are often highly fragmented, and the contiguous segments may have different bandwidths. When the number of contending users is large, it is particularly important to exploit every segment of the white spaces. The reduced bandwidth according to the user population offers an effective approach to fill up all segments of idle spectrum, even the narrow ones. Moreover, allocating the spectrum segments of different frequencies to the contending transmissions reduces the interferences among them. The allocated spectrum segments provide a natural separation for the concurrent transmissions. Compared with the fixed channelization scheme, the dynamic allocation adaptively creates the adequate number of channels to maximize the parallel transmissions and reduce the interferences in achieving high spectrum efficiency.

3. The fine control of bandwidth provides the sufficient number of channels and offers high accessibility to the spectrum. As a result, every flow obtains a timely transmission opportunity to exchange packets. A side effect is that the flow takes a smaller bandwidth of spectrum when the number of contending users is large. By adaptively adjusting the allocation, the dynamic spectrum allocation achieves better delay and jitter performance, which is preferable for TCP traffic. Through simulation studies in Section 3.5, we

demonstrate that the performance gain is up to 60% as compared to the fixed allocations.

### 2.3.3 Enabling Dynamic Spectrum Allocation

To enable the proposed concept of dynamic spectrum allocation, several supporting mechanisms are required.

First, the cognitive radios have to reliably detect the white spaces. However, a single scanner has limited detecting capability. It may miss a primary signal due to deep fading, shadowing effects, or existences of obstacles. Therefore, to realize the robust white space detection, it is important to combine the scanning results from the cognitive radios in the network. In this way, the chance of missing a primary signal can be largely reduced [93].

Second, since the dynamic spectrum allocation only decides the parameters of spectrum segments for communicating nodes, which node should access the spectrum segment is regulated by the MAC protocol. Therefore, it is critical to provide a scalable MAC design to support the dynamic spectrum allocation. In particular, when the number of contending users increases, the MAC protocol should *not* become the bottleneck of the system. It needs to be efficient to generate the access sequence to fill out the available spectrum, and delivers a stable performance in terms of the network throughput. Moreover, because the physical layers of client devices provide various and increasingly high transmission rates, it requires the MAC protocol to accommodate the different communications capabilities and exploit the rate diversities.

Third, the dynamic spectrum allocation decides the allocations depending upon the number of competing flows, the available spectrum and the prevailing allocations. Cognitive radios, therefore, need to collect such information to enable the dynamic spectrum allocation. The process of collecting the information

should be light-weighted, and provide up-to-date results. Furthermore, since the allocation separates the disjointed flows by assigning them different segment of spectrum, it poses challenges to maintain the connectivity in the network and to timely exchange allocation information.

Based on the proposed concept of dynamic spectrum allocation, we have built the cognitive radio systems for operating in white spaces, and the unlicensed bands. Chapter 3 and Chapter 4 describe the system designs for using white spaces and improving WLAN respectively. Each system includes the design of the supporting mechanisms described above. Through extensive simulations and analysis studies, we have shown the efficiency of adopting the proposed concept of dynamic spectrum allocation.

## 2.4 Chapter 2.3 In Nutshell

This chapter first explains the two recent trends in a great detail. First, white spaces in the TV bands are great resources for wireless data communications. The features of white spaces, including dynamic availability and fragmented frequencies, make existing wireless systems that are based upon fixed allocations, incapable of exploiting white spaces. Second, recent measurements of WLANs consistently show significant spatial and temporal variations of user and traffic load. The fixed channel structure and the IEEE 802.11 MAC offer very limited flexibility and scalability in handling such variations

To tackle these problems, we propose the dynamic spectrum allocation concept, and our hypothesis is that the dynamic spectrum allocation improves spectrum efficiency. Our proposed allocation scheme dynamically assigns to a wireless node an amount of spectrum as a function of total bandwidth of available spectrum, the number of contending nodes, and traffic load distribution in a lo-

cal network environment. It creates the appropriate number of channels to maximize potential parallel transmissions, and adjusts bandwidth assigned to each node for balancing traffic load variations across a network.

To support our dynamic allocation scheme, we exploit the advanced radio technology, so-called cognitive radios. These radios can sense the environment to identify primary operations, and adjust operating parameters to fill in unused spectrum. We have built the radio development board as the platform for enabling the dynamic spectrum allocation.

# Chapter 3

# KNOWS: Cognitive Networking Over White Spaces

*Science is a way of thinking much more than*
*it is a body of knowledge.*
*— Dr. Carl Sagan*
*The art and science of asking questions is*
*the source of all knowledge.*
*— Thomas Berger*

In this chapter, we present KNOWS, a cognitive wireless networking system, as the first part of our solution. KNOWS is a hardware-software platform that includes a spectrum-aware Medium Access Control (MAC) protocol and spectrum allocation algorithms. The KNOWS system improves the spectrum utilization by efficiently exploiting the white spaces in the licensed TV bands. We describe the system architecture and design of KNOWS in Section 3.1. We then in Section 3.2 describes the MAC protocol that enables nodes to reserve portions of the spectrum. A key challenge in the design of such a system is that of *Spectrum Allocation*, which enables nodes to reserve chunks of the spectrum for certain pe-

riods of time. In Section 3.3 we introduce the concept of *time-spectrum block* to model spectrum reservations, and use it to present a theoretical formalization of the spectrum allocation problem. We also present a centralized and a distributed protocol for spectrum allocation and show that these protocols are close to optimal in most scenarios. Section 3.4 and Section 3.5 present the analysis and simulation studies to demonstrate the effectiveness of KNOWS. We have implemented the distributed protocol in QualNet and show that our analysis closely matches the simulation results. We also show that in common scenarios KNOWS accomplishes a remarkable 200% throughput improvement over systems that use fixed allocation schemes. We discuss the alternative designs in Section 3.6 and compare the KNOWS design with other relevant schemes in Section 3.7. Finally, Section 3.8 summarizes the chapters and extracts the main contributions.

## 3.1 System Architecture and Design

The goal of the KNOWS system is to enable the dynamic spectrum allocation in cognitive radio networks. The nodes in KNOWS self-organize into a network without coordination from a central controller, and through the dynamic spectrum allocation, they maximize the overall spectrum utilization. This goal poses three main challenges that we address in our design.

- *Robust white space detection*: Unlicensed users need a robust way to discover the available white spaces. We note that different bandwidth chunks could be available at the sender and the receiver, therefore the goal is to use a spectrum chunk that is free for both of them.

- *Parallelism and connectivity*: There is a tradeoff between parallelism of flows and connectivity in the network. To enable parallelism, different flows should be active in different chunks of the spectrum. However, this might

prevent two nodes (users) that are part of the same network from communicating with each other. One approach to solve this problem is to use schemes that have been proposed by multi-channel MACs [76, 86]. However, these approaches incur extra overhead as described in Section 3.7.

- *Adaptive bandwidth selection*: The amount of bandwidth assigned to a pair of communicating nodes should depend on the total available spectrum, the contention intensity, and user traffic demand. Intuitively, when there are few users, each user should be assigned a wide bandwidth for a higher data rate; when there are more users in communication range, the total spectrum should be divided to accommodate more concurrent transmissions.

KNOWS addresses the above challenges as follows. First, KNOWS uses a collaborative scanning algorithm to detect incumbent operators in the TV bands. Therefore, only those portions of the spectrum that are detected to be available at all users are used for data communication. To address the second challenge, KNOWS uses a common signalling channel (in the ISM band) to maintain connectivity among nodes, even when they are transmitting or receiving on a different spectrum chunk. Parallelism is ensured by simultaneous data communication on the reconfigurable radio. KNOWS addresses the final challenge by allowing nodes to opportunistically use available spectrum resources by reserving chunks of bandwidth at a fine time-scale. The width of an allocated chunk depends on the amount of available spectrum and the number of contending nodes.

Figure 3.1 illustrates the architecture of KNOWS, which includes the hardware, the MAC (CMAC), and the spectrum allocation scheme (b-SMART). The hardware platform includes a dual-mode scanner and a reconfigurable radio. The scanner radio alternates between functioning as a scanner and a receiver. It scans the TV spectrum at least once every 30 minutes, as required by the FCC [74]. The scanner radio in our current platform takes less than 10 ms to scan one 6 MHz TV

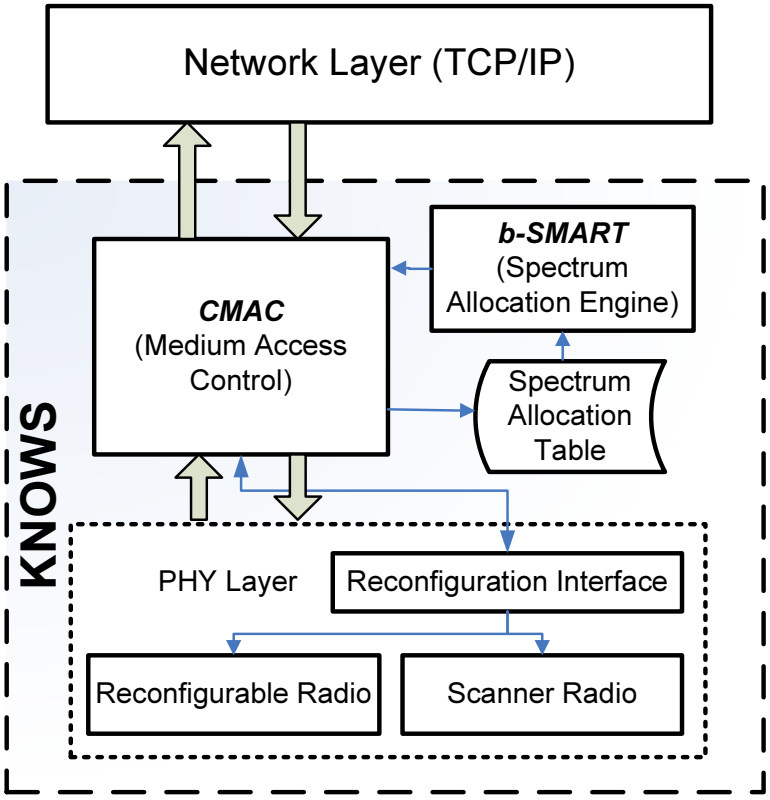


Figure 3.1: The components of KNOWS, which includes the hardware, a spectrum allocation engine and a MAC protocol.

channel. For most of the time, the scanner radio works as a receiver and is tuned to the 902–928 MHz unlicensed ISM band, which is used as a control channel.

To enable efficient spectrum sharing, each node stores the spectrum usage information in a local data structure, which we call the *spectrum allocation table*. The spectrum allocation engine in Figure 3.1 uses the spectrum allocation table to determine the portion of unused spectrum the node should reserve for its communication.

The spectrum allocation table records spectrum usage of neighboring unlicensed users in units of *time-spectrum blocks*. We define a time-spectrum block to be the time duration and the portion of spectrum that is reserved by a node for its communication. Figure 3.2 depicts one snapshot of time-spectrum block allocations stored in a spectrum allocation table. The bandwidth and time of the time-spectrum block is tuned according to the perceived contention intensity, the total available resources, and the queue length for each neighbor. The reconfigurable radio is then configured to operate in the defined time-spectrum block. It switches back to the control channel after the time-spectrum block is consumed.

Together with the scanner/receiver, CMAC builds the spectrum allocation table and implements a reservation-based mechanism that regulates spectrum access. Two communicating nodes first contend for spectrum access on the control channel. Upon winning contention, a handshake is performed, which enables b-SMART at the sender and the receiver to collaboratively agree on a time-spectrum block. The reservation is announced on the control channel to inform neighboring nodes. Accordingly, nodes populate their spectrum allocation tables with new reservations, and garbage collect the expired ones.

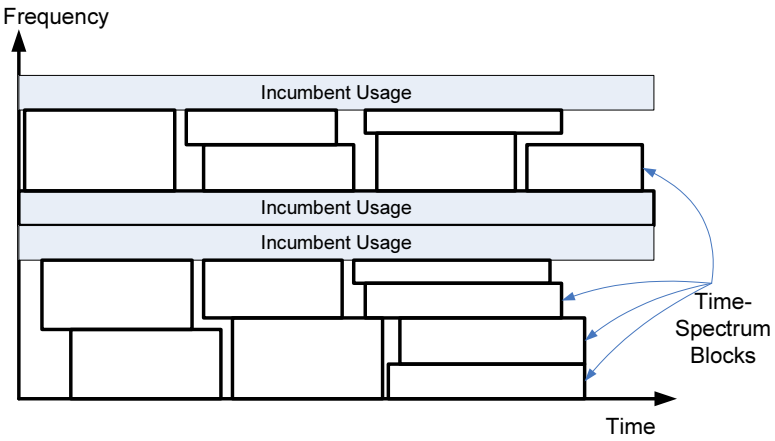


Figure 3.2: A snapshot of time-spectrum block allocations stored in a spectrum allocation table

## 3.2 CMAC: Spectrum Aware MAC

CMAC implements two main functions: it achieves collaborative sensing by combining scanning results in the one-hop neighborhood, and it realizes a spectrum reservation scheme using a common control channel.

When a node has packets to send it first contends for access to the control channel using CSMA/CA and random backoff mechanisms of IEEE 802.11 [40]. The pair of communicating nodes, upon winning access to the control channel, perform a three-way handshake. During the handshake process, the sender and the receiver exchange their local view of spectrum usage, decide on the spectrum block to use for the communication, and announce the reservation to their neighbors. On receiving a reservation packet, neighboring nodes store the reservation information in their local spectrum allocation table structure. At the start of the reserved time period, CMAC tunes the reconfigurable radio to the selected spectrum band and initiates the exchange of packets without any backoff.

### 3.2.1 Handshake

CMAC uses a three-way handshake, which builds on IEEE 802.11's two way RTS (request-to-send) and CTS (clear-to-send) handshake. In the handshake process, the senders contend for spectrum access on the control channel using the random backoff mechanism of IEEE 802.11. In particular, a transmitting node must first sense an idle channel for a time period of Distributed Inter-Frame Spacing (DIFS). After that it generates a timer chosen uniformly from the range $[0, w-1]$, where $w$ is referred to as the contention window. Initially, $w$ is set to $CW_{min}$. After each unsuccessful transmission, the value of $w$ is doubled, up to the maximum value $CW_{max} = 2^m CW_{min}$. The winning node sends a modified RTS packet to carry traffic load information and several proposed "time-spectrum blocks" to the receiver. A time-spectrum block is specified by the frequency interval $(f_0,\ f_0 + \Delta f)$ and the time interval $(t_0,\ t_0 + \Delta t)$. The regular control packets and our extended versions are shown in Figure 3.3. The modified RTS packet format incorporates the fields of queue length (1 byte) and average packet size (2 bytes) to describe the traffic load at the sender to the corresponding receiver. It also includes multiple time-spectrum blocks, each denoted by four fields: the starting frequency $f_0$ (1 byte), the bandwidth $\Delta f$ (1 byte), the start time $t_0$ (4 bytes), and the duration $\Delta t$ (2 bytes). The start frequency field records the offset value from the start frequency of the TV spectrum, which is 470 MHz in our system. We use 1 byte to denote frequency and bandwidth; this provides a resolution of 1 MHz. The start time and duration fields provide a timing resolution of one microsecond.

On receiving the RTS packet, the receiver chooses a time-spectrum block and informs the sender using a modified CTS packet after the Short Inter-frame Space (SIFS). The extended CTS packet contains address fields of the sender and the receiver, and details of the selected time-spectrum block. We introduce a new

control packet, DTS (Data Transmission reServation). The sender uses DTS to announce the spectrum reservation after receiving the CTS packet. CTS and DTS packets have the same format. Every node actively collects CTS and DTS packets to build the spectrum allocation table, which is a local view of spectrum usage in frequency and time. An entry in the structure corresponds to one reservation, denoted by the source and destination addresses and the time-spectrum block. The spectrum allocation table is updated each time the node receives a new CTS or DTS. The new entry is added to reflect the received reservation. Entries that expire or that share the same sender or receiver with the new entry are removed.

To reduce the time overhead caused by reservation nodes are allowed to make *advanced* reservations. Therefore, the handshakes are conducted in parallel with data transmissions. However, for design simplicity, each node is only allowed to have at most one valid outstanding reservation [56].

Note that the RTS packet can carry more than one time-spectrum block to convey more spectrum usage information to the receiver. However, the more information RTS carries, the higher will be the overhead on the control channel, and potentially higher loss rate. Our simulation study shows that using 1–2 time-spectrum blocks works best in most cases.

### 3.2.2 Data Transmission

A sender uses the reserved time-spectrum block to send data to the intended receiver. When a pair of communicating nodes switch to the selected segment of spectrum, they first perform physical-layer carrier sensing. If the selected spectrum is clear, nodes exchange packets without further back-off. Since the sender has exclusive access to the time-spectrum block, it can choose to transmit multiple packets back to back during the defined period.

Note that it is possible that after switching, the sender or the receiver find

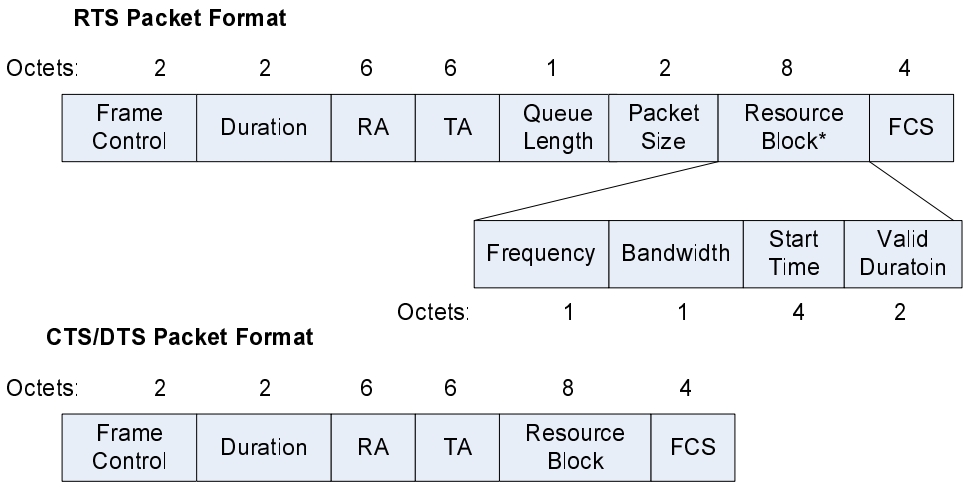


Figure 3.3: RTS/CTS/DTS packet format

the selected band to be busy; this may happen for three reasons:

1. The selected band may suffer from interference from transmissions in adjacent frequency bands.

2. The sender or the receiver may experience deep fading in the selected band, and/or

3. Conflicting reservations may occur due to loss of control packets.

If the sender or the receiver senses the selected band to be busy, it gives up the current time-spectrum block, and switches back to the control channel. If the other node does not sense the medium to be busy, and is unable to send or receive, it will wait for a pre-defined interval before switching back to the control channel.

### 3.2.3 Collaborative Sensing

Ideally, there should be no communication by unlicensed users in a spectrum chunk when it is being scanned for TV signals. This increases the accuracy of a scan reported by KNOWS. To achive this goal, i.e. prevent a neighbor

from sending a packet in the spectrum chunk being scanned, a node reserves the spectrum it wants to scan by sending a DTS packet with this information. Therefore, other nodes do not send a packet in that spectrum chunk, while it is being scanned by another node. During the scanning period, the reconfigurable radio of the node resides on the control channel, and collects control packets until the scanner finishes its task.

Collaborative sensing in KNOWS has two components. First, it creates a quiet period (i.e., not in use by peer devices in the vicinity) for a node to sense a segment of the TV spectrum. To do so, the scanning node sends a DTS packet to reserve the spectrum segment. When the reserved time arrives, the scanner scans the selected spectrum. During the scanning period, the reconfigurable radio resides in the control channel and collects control packets until the scanner finishes the task.

CMAC aggregates scanning results from neighbors for better detection of incumbent operators. It extends the beacon frame format used in 802.11 with a bitmap field that carries local scanning results. A bit represents the occupancy status of the corresponding TV channel, with 0 for occupied and 1 for empty. In our case, CMAC can use up to 30 TV channels from channel 21 to channel 51 (except channel 37) [74]. The bitmap field, therefore, requires 4 bytes to reflect the activities in the UHF spectrum. Each node transmits the beacon message every beacon interval (typically 100–200 ms). A node receiving a beacon message updates the stored scanning results by applying a bit-wise AND operation.

### 3.2.4 Time Synchronization

CMAC performs spectrum allocation via a reservation-based mechanism; hence it is critical to ensure that the nodes have synchronized clocks. The timestamp field in the aforementioned beacon message is used for time synchroniza-

tion. The timestamp field is 64 bits and offers microsecond time resolution. To synchronize the system clock, upon receiving the beacon, the node records its local time, $T_L$, extracts the timestamp field, $T_B$, estimates the transmission time of the beacon, $t$, and then synchronizes its system clock by adding $(T_B + t) - T_L$. To avoid cyclic synchronization, nodes only synchronize to the faster clock in the system. The accuracy of time synchronization is affected by the interrupt handler and the estimation of propagation delay, which vary across different systems. In our system, nodes make reservations only with their one-hop neighbors. Therefore, we need to focus on the clock skew among one-hop neighbors. As shown in [44, 54], the clock skew among nodes within one hop can be controlled to less than 1 $\mu s$ using beacon messages.

### 3.2.5 Bootstrap

To join the network, a new node performs the following operations. It first uses the scanner to generate a list of unused TV channels, and at the same time tunes the reconfigurable radio to the control channel, waiting for beacon messages from other nodes for time synchronization. After scanning completes, the node combines the scanning results obtained from beacons, and sends out a beacon frame every beacon interval.

### 3.2.6 Other Considerations

In order to avoid the control channel from becoming a bottleneck and to ensure fairness among contending nodes, appropriately adjusting the random back-off mechanism is crucial. In particular, it is well-known that even in the basic 802.11 protocol and even for a single communication channel, the problem of setting the back-off window to its minimum size upon completion of a transmis-

sion causes unfairness. A node $v$ that has just finished transmitting has a higher chance of recapturing the channel than other nodes, because $v$'s back-off window is smaller and hence, its probability of sending an RTS is higher.

Whereas this fundamental problem has a relatively minor impact in single-channel scenarios, it becomes much more dramatic in multi-channel settings or cognitive radio networks. In such scenarios, a large number of nodes can simultaneously transmit data on a band other than the control channel and hence, the rate of nodes returning to the control channel is significantly higher than in single-channel environments. If each of these returning nodes sets its contention window to the minimum size, other nodes have virtually no chance of gaining access to the channel and starve.

In order to alleviate this problem, our protocol implements the following enhancement to the 802.11-based back-off protocol: *When returning to the control channel upon completion of a transmission, a node chooses the smallest possible back-off window size larger than $N$, i.e., the smallest window for which its transmission probability is smaller than $1/N$.* This improvement of the protocol prevents an over-increase of the sum of transmission probabilities of all nodes, which, in turn, guarantees that the *control channel does not become a bottleneck* even when the number of nodes $n$ becomes very large and significantly exceeds the maximum number of available channels. This is also backed by our analysis in Section 3.4.

## 3.3 Dynamic Spectrum Allocation in White Spaces

A key challenge in the KNOWS design is to implement the dynamic spectrum allocation, which enables cognitive radio nodes to reserve chucks of the spectrum for a certain period of time. In this section, we elaborate the concept of a *time-spectrum block* for modeling spectrum spectrum and develop a comprehen-

sive framework to study the problem of spectrum allocation over white spaces. We first use time-spectrum blocks to formulate a theoretical problem of the spectrum allocation in cognitive radio networks in Section 3.3.1. We present a centralized and a distributed protocol for spectrum allocation in Section 3.3.3 and Section 3.3.4 respectively. Section 3.4 and Section 3.5 show that these protocols achieve the performance close to optimal in most scenarios.

### 3.3.1 Problem Formulation

Our problem formulation for dynamic spectrum allocation in a cognitive radio is based on the radio platform we describe in Section 2.2. The prototype consists of a reconfigurable transceiver and a scanner. The parameters of the transceiver, such as the center frequency and bandwidth, can be adjusted within 10s of microseconds. The transceiver can only tune to a *contiguous* segment of spectrum. Due to hardware limitations, the possible bandwidth values are a discrete set in the range of [$b_{min}$, $b_{max}$], where $b_{min}$ and $b_{max}$ denote the lower and upper bounds of the supported bandwidth, respectively. The largest usable bandwidth is typically below 40 MHz. We are also including an extra receiver radio in the prototype to enable a common control channel for exchanging spectrum usage information. In our previous discussion 2.2, we have shown that the extra receiver can be implemented using the scanner radio.

While our model and problem formulation is specifically tuned to capture the physical capabilities of our KNOWS prototype, it is general enough to allow for a wide range of analytical and algorithmic studies of spectrum allocation problems in cognitive radio networks. The model is primarily intended to capture the novel algorithmic challenges arising from the nodes's capability to adaptively change their bandwidth (in addition to their center-frequency).

## Definitions and Notation:

We model the cognitive radio network as a set of $n$ nodes $V = \{v_1, \ldots, v_n\}$ located in the two-dimensional Euclidean plane. Let $d(v_i, v_j)$ denote the Euclidean distance between $v_i$ and $v_j$. Let $f_{bot}$ and $f_{top}$ denote the lower and upper end of the accessible target spectrum, e.g., $f_{bot} = 470MHz$ and $f_{top} = 698MHz$ in the TV spectrum [31]. Each node $v_i \in V$ is equipped with a radio transceiver that is capable of dynamically accessing any *contiguous* frequency band $[f, f + \Delta f]$ for all $f_{bot} \leq f \leq f + \Delta f \leq f_{top}$, as long as $b_{min} \leq \Delta f \leq b_{max}$.

For each pair of nodes $(v_i, v_j) \in V$ within mutual communication range, $D_{ij}(t, \Delta t)$ denotes the *demand* in bit/s that $v_i$ would like to transmit to $v_j$ during time interval $[t, t + \Delta t]$. This link-based demand subsumes the traffic of all flows that are routed over this particular link[1] and our definition captures the fact that demands may vary both between different links and also on a single link over time.

The crucial difference between the spectrum access using predefined channels of fixed channel-width and the dynamic access in cognitive radio networks is that (i) the *bandwidth* (channel-width) of the spectrum allocated to different links becomes an additional variable, and (ii) the radio parameters can be adjusted in a fine timescale. We say that a *time-spectrum block* $B_{ij}^k = (t_k, \Delta t_k, f_k, \Delta f_k)$ is assigned to link $(v_i, v_j)$ if sender $v_i$ is assigned the contiguous frequency band $[f_k, f_k + \Delta f_k]$ of bandwidth $\Delta f_k$ during time interval $[t_k, t_k + \Delta t_k]$. We can therefore view the dynamic spectrum allocation problem as dynamic packing of time-spectrum blocks into a three-dimensional resource, consisting of time, frequency, and space. Suppose node $v_i \in V$ transmits to receiver $v_j$ using a time-spectrum block $B_{ij}^k$. The

[1] Our definition essentially describes a link-based notion of scheduling. All definitions and theoretical results in this paper can easily be extended to broadcast scheduling problems, in which demands are determined *per node*, instead of per link.

amount of data $v_j$ receives in $[t_k, t_k + \Delta t_k]$ can be expressed as [49]:

$$C_{ij}(B_{ij}^k) = \Delta f_k \cdot \mathcal{A}(f_k, \Delta f_k) \cdot \Delta t_k \cdot \mathcal{B}(\Delta t_k). \tag{3.1}$$

The function $\mathcal{A}(f_k, \Delta f_k)$ characterizes how well the band $[f_k, f_k + \Delta f_k]$ can be utilized, which depends on the frequency, the bandwidth, the spectrum condition as well as on the hardware. The function $\mathcal{B}(\Delta t_k)$ captures the hardware and protocol-specific overhead incurred when accessing the spectrum (for example the overhead incurred by contentions and sending acknowledgements). We call $C_{ij}(B_{ij}^k)$ the *capacity* of the allocated time-spectrum block. Under ideal channel conditions and disregarding any potential overhead, the "ideal" capacity simplifies to $C_{ij}(B_{ij}^k) = \gamma \Delta f_k \Delta t_k$ for some constant $\gamma$. However, this definition is oversimplified in the sense that it allows for an overhead-free slicing of time into infinitely fine-grained blocks. Therefore, in this paper, we study a more realistic capacity definition that precludes this possibility. We assume spectrum utilization to be linear in the bandwidth, but $\mathcal{B}(\Delta t_k) = (1 + \beta/\Delta t_k)$, i.e.,

$$C_{ij}(B_{ij}^k) = \alpha \Delta f_k (\Delta t_k - \beta), \tag{3.2}$$

for a constant $\beta$, that represents the overhead incurred when accessing the spectrum band. This overhead may include the time overhead of switching frequency or the time used for medium access contention.

### Interference Model:

In principle, the *dynamic spectrum allocation problem* can be analyzed using a variety of underlying network and communication models, for instance the classic *protocol* and *physical models* [38]. In this paper, we consider a cognitive radio in-

terference model based on the simple protocol model.[2] In this model, each sender $v_i$ is associated with a transmission range $R_i^t$ and a (larger) interference range $R_i^{int}$. A message sent over a link $(v_i, v_j)$ is possible if there is no simultaneous transmitter $v_z$ such that $v_j$ is in $v_z$'s interference range $R_z^{int}$. That is, two time-spectrum blocks $B_{ij}^k(t_k, \Delta t_k, f_k, \Delta f_k)$ and $B_{gh}^\ell(t_\ell, \Delta t_\ell, f_\ell, \Delta f_\ell)$ are mutually *non-interfering* if one of the following conditions is satisfied.

- $d(v_j, v_g) > R_j^{int}$ and $d(v_i, v_h) > R_i^{int}$ (space separation)
- $\max\{f_k, f_\ell\} \geq \min\{f_k + \Delta f_k, f_\ell + \Delta f_\ell\}$ (freq. separation)
- $\max\{t_k, t_\ell\} \geq \min\{t_k + \Delta t_k, t_\ell + \Delta t_\ell\}$ (time separation)

Since a cognitive radio incorporates a scanner to detect primary signals, mitigating interference among secondary users is the key challenge facing the dynamic spectrum allocation. We define a set of *prohibited bands* $\mathcal{P} = \{P_1, \ldots, P_L\}$, where every $P_\ell \in \mathcal{P}$ denotes a spectrum band $P_\ell = [f_y, f_z]$ that is used by a primary station and detected by the scanner. A *spectrum allocation schedule* $\mathcal{S}$ is an assignment of time-spectrum blocks $B_{ij}^k$ to links $(v_i, v_j) \in E$, such that no two assigned blocks $B_{ij}^k, B_{gh}^\ell \in \mathcal{S}$ interfere and no prohibited spectrum is used. Formally, a schedule is $S$ *feasible* if the following conditions hold.

- No two assigned time-spectrum blocks interfere
- $[f_i, f_i + \Delta f_i] \cap P_\ell = \emptyset$ for every assigned block $B_{ij}^k$ and every $P_\ell \in \mathcal{P}$.

**Dynamic Spectrum Allocation Problem:**

Equipped with these definitions, we now state the dynamic spectrum allocation problem.

[2] Even in classic single-channel networks, the protocol and physical models allow for vastly different communication patterns [70] if transmission powers very between nodes. In case of uniform transmission powers, however, the two models exhibit similar characteristics. Studying the spectrum allocation problem with varying transmission powers significantly adds to its complexity and is an interesting avenue for future research.

*Given dynamic demands $D_{ij}(t_k, \Delta t_k)$, a dynamic spectrum allocation protocol computes a feasible spectrum allocation schedule $\mathcal{S}$ that assigns non-interfering dynamic time-spectrum blocks to links such that demands are satisfied as much as possible, i.e., the spectrum is efficiently utilized.*

Numerous specific measures and combinatorial optimization problems can be derived from the above formulation. The first such measure of interest that characterizes the performance of protocols is *throughput*. As $C_{ij}(B_{ij}^k)$ is the maximum amount of data that can be sent over link $(v_i, v_j)$ in time-spectrum block $B_{ij}^k$, the throughput of link $(v_i, v_j)$ in $[t_k, t_k + \Delta t_k]$ is

$$T_{ij}(B_{ij}^k) = \min\{D_{ij}(t_k, \Delta t_k), C_{ij}(B_{ij}^k)\} \tag{3.3}$$

and the *throughput maximization problem* asks for a feasible schedule $\mathcal{S}$ that maximizes

$$T_{max} = \sum_{(v_i,v_j)\in E} \sum_k T_{ij}(B_{ij}^k).$$

As the throughput measure does not account for any notion of fairness, we want to maintain *proportionally-fair throughput* among different demands. For some demand $D_{ij}(t_k, \Delta t_k)$, let $\mathcal{I}_{ij}$ denote all time intervals $[t, t+\omega]$ for some fixed duration $\omega$, for which $[t, t+\omega] \in [t_k, t_k+\Delta t_k]$. Then, the *minimum proportionally-fair throughput* $T_{minfair}(\omega)$ is

$$T_{minfair}(\omega) = \min_{(v_i,v_j)\in E} \min_{[t,t+\omega]\in\mathcal{I}_{ij}} \frac{T_{ij}(t, t+\omega)}{\omega \cdot D_{ij}(t_k, \Delta t_k)},$$

where $T_{ij}(t, t+\omega)$ is the throughput achieved during interval $[t, t+\omega]$. A high minimum proportionally-fair throughput therefore guarantees that in every time-interval of length $\omega$, every demand gets its fair share of throughput. The shorter $\omega$ is chosen, the more short-term and fine-grained this notion of fairness be-

comes. In particular, a protocol that guarantees good minimum proportionally-fair throughput for very small values of $\omega$ (say, in the order of a few milliseconds) leads to low latencies and minimizes jitter.

### 3.3.2 Problem Complexity Analysis

As it turns out, even simple and highly restricted variants of the dynamic spectrum allocation problem are computationally hard. In particular, the throughput problems are hard even if demands have infinite duration (i.e., $\Delta t = \infty$), and sometimes even in single-hop scenarios or when there are no prohibited bands, i.e., $|\mathcal{P}| = 0$.

**Theorem 3.3.1.** *The proportionally-fair throughput maximization problem is NP-complete even if* $|\mathcal{P}| = 0$.

*Proof.* The proof is by reduction to the NP-complete 3-chromatic number problem in unit disk graphs, which is to determine whether a given unit disk graph has a feasible vertex coloring using at most $3$ colors [37]. More precisely, we consider a slightly more restricted family of unit disk graphs in which every node appears in a clique of size $3$. The proof of [37] can be adjusted to show that the 3-chromatic number problem remains hard in such graphs. The reduction works as follows: Given an instance of the 3-chromatic number problem in unit disk graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ in which every node appears in a clique of size $3$, we create the following instance of the dynamic spectrum allocation problem: For each node $v_i \in \mathcal{V}$, there are two cognitive radio nodes, a sender $w_i$ and a receiver $w_i'$. Node locations of nodes $w_i$ in the plane correspond to the positions of $v_i$ in $V$, but are scaled in such a way that $R_i^{int}$ corresponds to the unit distance of $\mathcal{G}$. Finally, each node $w_i'$ is placed such a way that it sees the same set of interfering nodes as $w_i$, i.e., for all

$w_j, i \neq j$,

$$d(w_i, w_j) \leq R_j^{int} \iff d(w_i', w_j) \leq R_j^{int}.$$

A close inspection of the hard instances derived in [37] reveals that finding such a placement is always possible. Finally, let the demand $D_{ii}'$ of $(w_i, w_i')$ be $D_{ii}' = (f_{top} - f_{bot})/3$, and all other demands $D_{ij} = 0$. All demands are invariable in time. It holds that $\mathcal{G}$ is 3-colorable exactly if the maximum total throughput is $T_{max} = f_{top} - f_{bot}$. In particular, if $\mathcal{G}$ is 3-colorable, each sender $w_i$ with color $c_i$ in the original graph is assigned a time-spectrum block $(0, \infty, f_{bot} + (c_i - 1)(f_{top} - f_{bot})/3, (f_{top} - f_{bot})/3)$. By definition of the reduction to the coloring problem and the placement of receivers $w_i'$, all these blocks are non-interfering and hence, every demand has a proportional throughput of exactly $T_{ij}/D_{ij} = 1 - \beta/\Delta t$. On the other hand, if $\mathcal{G}$ is not 3-colorable, every solution to the dynamic spectrum allocation problem assigns a time-spectrum block of bandwidth at most $1/4$ to at least one link at every time. Hence, for at least one node, $T_{ij}/D_{ij} > 1 - \beta/\Delta t$. Unless $P = NP$, no efficient algorithm can distinguish these two cases, which concludes the proof. □

The following theorem can be proven analogously.

**Theorem 3.3.2.** *The total throughput maximization problem is NP-complete even if* $|\mathcal{P}| = 0$.

Finally, the problem becomes NP-hard even in single-hop networks if there are forbidden spectrum bands.

**Theorem 3.3.3.** *For any* $|\mathcal{P}| > 0$, *the proportionally-fair throughput maximization problem is NP-complete even in a single-hop environment.*

*Proof.* This proof follows by reduction to the PARTITION problem. Given a set $A$ of numbers $a_i$, the question is whether $A$ can be partitioned into two sets

$A_1, A_2$ in such a way that $\sum_{i \in A_1} a_i = \sum_{i \in A_2} a_i = \frac{1}{2} \sum_{i \in A} a_i$. Given an instance of the PARTITION problem, define a single-hop instance of the dynamic spectrum allocation problem such that each value $a_i$ corresponds to two nodes $w_i, w_i'$ with demand $D'_{ii} = a_i \cdot (f_{top} - f_{bot} + x) / \sum_{i \in A} a_i$. Additionally, let $\mathcal{P} = P$ with $P = (f_{top} - f_{bot} - x)/2, (f_{top} - f_{bot} + x)/2$. In this setting, it is easy to see that an optimally porportionally fair allocation is possible if and only if the set of demands can be divided into two equal partitions. □

### 3.3.3 A Centralized Allocation Scheme

In order to shed light into the fundamental nature of the problem, we first study a simple centralized algorithm whose performance with regard to (short-term) proportionally-fair throughput is provably good even in worst-case networks. Based on our studies, we derive three desirable properties of a practical, distributed solution, which we investigate in subsequent sections.

The centralized algorithm assume that each node only has one outgoing demand and that $|\mathcal{P}| = 0$. Without loss of generality, we also assume that $f_{bot} = 0$. For notation, let $\Delta_{min}$ be the minimum duration of any demand and let $\chi = \Delta_{min}/\beta$ be the ratio between the minimum demand's duration and the switching overhead time. In practice, $\chi$ is a large constant. Let $D_{ij}(t)$ denote the current demand of link $(v_i, v_j)$ at time $t$, and let $S_{ij}(t)$ be the sum of demands of all links in $E_{ij}$ at time $t$. For a link $(v_i, v_j)$, we denote by $E_{ij}$ the set of links that cannot be scheduled together with $(v_i, v_j)$ at the same time using the same frequency band. According to our definition, a solution which assigns to each node blocks of bandwidth $D_{ij}(t)(f_{top} - f_{bot})/S_{ij}(t)$ is proportionally-fair at time $t$.

In a nutshell, Algorithm 1 works as follows. Periodically, after a time-interval of size $\Gamma = \chi\beta/k$, it attempts to readjust the current spectrum assignment and assigns new time-spectrum blocks to each active link. The idea is that, on

**Algorithm 1** Centralized Spectrum Allocation Algorithm

1: Define the constants $\Gamma = \chi\beta/k$ for some $3 \leq k < \chi$.
2: Schedule at time $t_{cur}$:
3: Let $A_{cur} = \{D_{ij}(t, \Delta t) \mid t + \Delta t \geq t_{cur} + \Gamma\}$ be the set of *active demands*.
4: Let $D'_{ij} = \min\{2^i, i \in \mathbb{Z} | 2^i \geq D_{ij}(t, \Delta t)\}$ for each demand in $A_{cur}$.
5: **for each** $D_{ij}(t,\Delta t) \in A_{cur}$ in nonincreasing order of $D'_{ij}$:
6: Let $\mathcal{I}_{cur}$ be the set of intervals already assigned at time $t_{cur}$. For $D_{ij}(t, \Delta t)$, assign a frequency interval $I_{ij} = [\ell_{ij}, \ell_{ij} + D'_{ij}]$ such that it does not overlap with any previously assigned interval in $\mathcal{I}_{cur}$. By definition of $D'_{ij}$, such an interval always exists and can be found easily.
7: **end for**
8: Let $\Phi_{max}$ be the highest upper boundary $\ell_{ij} + D'_{ij}$ of any interval $I_{ij}$ assigned to any active demand in $A_{cur}$.
9: Set $\varphi = (f_{top} - f_{bot})/\Phi_{max}$ and assign to each link $(v_i, v_j)$ with active demand the time-spectrum block $B_{ij}^{cur} = (t_{cur}, \Gamma, \varphi\ell_{ij}, \varphi D'_{ij})$.
10: $t_{cur} = t_{cur} + \Gamma$.

the one hand, the algorithm should adjust quickly enough to response in the demand variation, but on the other hand, too frequent reallocation of time-spectrum blocks is inefficient due to the overhead time $\beta$. The definition of $\Gamma$ ensures a good balance between these two contradictory aims.

Within a single time-interval, the algorithm tries to maximize proportionally-fair spectrum usage by greedily assigning frequency intervals to nodes with *active demands*. A demand is called *active* if its duration spans the entire time-interval $[t_{cur}, t_{cur} + \Gamma]$. Particularly, demands are rounded to the next higher power of 2. In non-increasing order of this demand-size, frequency-intervals $I_{ij}$ are then allocated to links with active demands in a simple greedy fashion. The underlying reason for thus rounding demands in Line 4 is that, in combination with the greedy allocation of bandwidths, this *avoids fragmentation* of the spectrum (cf the proof of Lemma 3.3.5) even in multi-hop scenarios. This guaranteed absence of

unwanted fragmentation in the greedy-allocation phase is the key design principle of the algorithm.

The initial allocation typically leads to an infeasible solution, and in order to fix this, all assigned frequency intervals are linearly scaled down by a factor of $\varphi = (f_{top} - f_{bot})/\Phi_{max}$, where $\Phi_{max}$ is a scaling factor that ensures feasibility. We now show that our algorithm is within a constant ratio of the optimal solution with regard to $T_{minfair}(\Delta)$ for any value between $3\beta \leq \Delta \leq \chi\beta$.

**Lemma 3.3.4.** *All assigned time-spectrum blocks in the resulting schedule are mutually non-interfering.*

*Proof.* Blocks allocated in different time-intervals $[t, t+\Gamma]$ clearly do not interfere. Within a given $[t, t+\Gamma]$, Line 6 guarantees that the intervals $I_{ij}$ are non-interfering. As all links use the same scaling factor $\varphi$, the lemma follows. □

The next lemma states that in each time-interval $[t, t+\Gamma]$, every active demand receives a time-spectrum block whose bandwidth is close to the optimal proportionally-fair one.

**Lemma 3.3.5.** *Let $\kappa$ be a constant. Let $OPT_{ij}(t_{cur}, \Gamma)$ denote the (minimal) throughput of $(v_i, v_j)$ in $[t_{cur}, t_{cur} + \Gamma]$ in an optimally proportionally-fair solution. For every link $(v_i, v_j)$ with active demand $D_{ij}(t, \Delta t) \in A_{cur}$, it holds that*

$$T_{ij}(t_{cur}, \Gamma) \geq \frac{\kappa}{2} \cdot OPT_{ij}(t_{cur}, \Gamma) \cdot \frac{\Gamma - \beta}{\Gamma}.$$

*Proof.* Each time a node is assigned a new time-spectrum block, it incurs an overhead of time $\beta$. It follows that for an active node,

$$T_{ij}(t_{cur}, \Gamma) \geq \varphi D'_{ij}(\Gamma - \beta) = \frac{D'_{ij}(f_{top} - f_{bot})}{\Phi_{max}}(\Gamma - \beta). \tag{3.4}$$

A key ingredient for the proof is now that

$$\Phi_{max} \leq 2S_{gh}(t) \tag{3.5}$$

for the neighborhood $E_{gh}$ with maximal $S_{gh}(t)$. This inequality is true because in Line 4, every demand is increased by at most a factor of $2$ and—crucially—there is *no fragmentation*: Because demands are all powers of $2$, and time-spectrum blocks are assigned greedily in non-increasing order, it holds that when a demand $D'_{ij}$ is assigned, every available non-overlapping free space has size at least $D'_{ij}$. That is, in the neighborhood of link $(v_g, v_h)$ with maximal $S_{gh}(t)$, the only reason why $\Phi_{max}$ may be larger than $S_{gh}(t)$ is the rounding.

Furthermore, it holds that in each time-window $[t_{cur}, t_{cur}+\Gamma]$ only active demands are considered, i.e., demands that span the entire interval. By standard geometric arguments that hold in the protocol model as long as $R_i^{int}/R_i^t = O(1)$ [65], we obtain for some constant $\kappa \in O(1)$ that

$$OPT_{ij}(t_{cur}, \Gamma) \;\leq\; \frac{1}{\kappa} \cdot \Gamma \cdot \frac{f_{top} - f_{bot}}{S_{gh}(t)}. \tag{3.6}$$

Plugging (3.5) and (3.6) into inequality (3.4) concludes the proof. □

Finally, we are ready to derive the theorem. The *volatility-ratio* $\Psi$ denotes the largest possible ratio between minimum and maximum $S_{ij}(t)$ of any link $(v_i, v_j) \in E$ during a time interval of duration $\beta$.

**Theorem 3.3.6.** *In every network and for every $3 \leq k < \chi$, Algorithm 1 is within a factor of $O\left(\Psi \cdot \frac{\chi}{\chi - k}\right)$ of the optimal solution with regard to $T_{minfair}(\Delta)$, where $\Delta = 3\chi\beta/k$.*

*Proof.* By the definition of $\chi < \Delta_{min}/\beta$ and $\Gamma = \chi\beta/k$, we know that every demand $d_{ij}(t, \Delta t)$ appears in at least three consecutive time-windows $[t_{cur}, t_{cur} +$

$\Gamma]$, at least one of which it spans entirely. In this time-interval, $D_{ij}(t, \Delta t)$ becomes *active* and therefore, Lemma 3.3.5 implies that the throughput $T_{ij}(t_{cur}, \Gamma)$ is close to the optimum. The first and last time-window $D_{ij}(t, \Delta t)$ appears in, it is not scheduled by the algorithm, but an optimal schedule may have assigned a time-spectrum block to this link. By the definition of volatility, we know that $OPT_{ij}(t_{cur}, 2\Gamma) \leq (\Psi + 1)OPT_{ij}(t_{cur}, \Gamma)$ and, similarly, $OPT_{ij}(t_{cur} - \Gamma, 2\Gamma) \leq (\Psi + 1)OPT_{ij}(t_{cur}, \Gamma)$. Because this holds for every $t_{cur}$, it follows from Lemma 3.3.5,

$$T_{ij}(t_{cur}, 3\Gamma) \geq \frac{\kappa}{2(\Psi + 1)} \cdot OPT_{ij}(t_{cur}, 3\Gamma) \cdot \left(1 - \frac{\beta}{\Gamma}\right).$$

The proof of the theorem is now concluded by deriving the ratio $OPT_{ij}(t_{cur}, 3\Gamma)/T_{ij}(t_{cur}, 3\Gamma)$ and replacing $\beta/\Gamma$ by $k/\chi$. □

Notice that the bound in Theorem 3.3.6 gets tighter as $k$ increases. That is, the smaller the fairness interval, the looser the proportionally-fair throughput guarantee. Setting $k = \chi/2$, for instance, we obtain an $O(\Psi)$ approximation of the optimum for an interval as small as $6\beta$.

### 3.3.4 b-SMART: A Distributed Scheme

The centralized algorithm discussed in this section does not lend itself for practical application in real multi-hop cognitive radio networks. The purpose of our studies has been to gain a deeper understanding of the inherent algorithmic challenges that a practical protocol for the dynamic spectrum allocation problem faces, and consequently, what desirable properties such a protocol should have.

*Opportunistic usage:* Spectrum allocation should divide the overall bandwidth $B$ of white spaces to accommodate the contending links by tuning the operating bandwidth. In the centralized solution, the bandwidth assigned to the transmission is $D_{ij}B/S_{ij}$, which allows the spectrum to be adaptively bundled

together to deliver high throughput for heavy traffic from few users, or be opportunistically fragmented and shared among a large number of contending devices.

*Fine-timescale control:* Links should share the spectrum in fine timescale in order to adapt to instantaneous traffic demand and control latency. $\Gamma = \chi\beta/k$ is defined to achieve a balance between the agility of solutions and the time overhead associated with each allocation.

*Non-interfering allocation:* All time-spectrum blocks are mutually non-interfering. The exclusive access to a time-spectrum block largely reduces time overhead of contentions in the given band, mitigates the hidden terminal problem after frequency switching [86], and encourages packet aggregation for high efficiency.

We now present b-SMART, a distributed and practical spectrum allocation scheme. In order to realize dynamic and fine-timescale allocation, nodes running the b-SMART protocol maintain the instantaneous spectrum usages of all their neighbors. The sender and the intended receiver coordinate with each other to reserve a time-spectrum block that is available at both nodes. The size of the block is determined using a local algorithm that is executed at the sender and the receiver.

The CMAC scheme discussed previously only regulates which sender-receiver pair may reserve some time-spectrum block at a specific time. At the heart of our protocol is the dynamic spectrum allocation algorithm that decides on the four variables $t$, $\Delta t$, $f$, and $\Delta f$ to shape the block. The algorithm is invoked when a sender sends a RTS packet to an eligible receiver. The sender considers the overall bandwidth of the white spaces $B = f_{top} - f_{min} - |\mathcal{P}|$, the local spectrum allocation table, and the corresponding queue size $qLen$, to decide on a time-spectrum block.

This dynamic decision is guided by the following principles, which are derived in Section 3.3.3. First the bandwidth $\Delta f$ should be determined based on

**Algorithm 2** Distributed Spectrum Allocation Algorithm

Let the available bandwidth options be: $\{b_1, b_2, ..., b_n\}$, with $b_1 < b_2 < \ldots < b_n$.
$I := \min\{i | b_i \geq B/N\}$;
**for** $i = I, \ldots, 1$ **do**
  $\Delta f = b_i; \Delta t := \min\{T_{max}, \text{Tx\_Duration}(b_i, qLen)\}$
  **if** $\Delta t == T_{max}$ or $i == 1$ **then**
    Find the best placement of the $\Delta f \times \Delta t$ block in the local spectrum allocation table that minimizes the finishing time and is compatible with the existing allocations and prohibited bands.
    **if** the block can be placed in the local spectrum allocation table **then**
      **return** the allocation $(t, \Delta t, f, \Delta f)$.
    **end if**
  **end if**
**end for**

the current demand; it must be large enough to achieve a high data-rate, but it should not be too large considering the fragmentation in the white spaces and the fairness expectation. Hence, b-SMART attempts to assign to each sender-receiver pair $(v_i, v_j)$ a time-spectrum blocks with bandwidth $B/N$, where we define $N$ is the number of current disjoint transmissions in the interference range of $(v_i, v_j)$. Two transmissions are considered disjoint if they do not share either endpoint. This definition of $N$ aims at achieving *per-node fairness* and is particularly appealing because it can easily be implemented in the distributed setting. The second design trade-off involved is the block duration $\Delta t$. While using a shorter block reduces delay and improves connectivity, it results in higher contention on the control channel. Our approach sets an upper bound $T_{max}$ on the maximum block-duration and nodes always try to send for duration $T_{max}$. As we motivate later, our choice of $T_{max}$ amortizes the incurred overhead in the control channel, thereby preventing the control channel from becoming a bottleneck.

The details of this algorithm are presented in Algorithm 2. The algorithm evaluates the available bandwidth options in decreasing order, starting with the bandwidth option just exceeding $B/N$. For each bandwidth option $\Delta f = b_i$ in consideration, the algorithm estimates its corresponding transmission time $\Delta t$

based on the current queue size (i.e., how long it uses the specific bandwidth to empty the queue). If the resulting transmission time exceeds $T_{max}$, then $\Delta t$ is set to $T_{max}$. Given $\Delta t$ and $\Delta f$, we optimize the placement of the time-spectrum block in the local allocation table. This is done by minimizing the finishing time while not overlapping with the existing allocations and prohibited bands. If the time-spectrum block of size $\Delta t \times \Delta f$ cannot be placed due to the existence of prohibited bands, then the next smaller bandwidth option is considered.

We next discuss our choice for the two important parameters, $T_{max}$ and $N$, which greatly influences the efficiency and accuracy of b-SMART .

**Setting $T_{max}$:** Let $\Lambda \triangleq B/b_{min}$ denote the maximum number of parallel transmissions that the white spaces of bandwidth $B$ can accommodate, where $b_{min}$ is the smallest bandwidth option. In b-SMART , $T_{max}$ is set to satisfy the following condition:

$$T_{max} = \Lambda \cdot T_o = B \cdot T_o / b_{min}, \tag{3.7}$$

where $T_o$ denotes the average time spent on one successful handshake in the control channel. Before validating our choice by means of analysis and simulation in Sections 3.4, we briefly explain the main intuition behind the formula. In order to keep the white spaces fully utilized, we need to prevent the control channel from becoming the bottleneck. Therefore, it is important to ensure that the rate at which handshakes are generated, $R_l$, is not less than the rate at which nodes return to the control channel, $R_r$, i.e., $R_l \geq R_r$. The handshake is generated at the rate of $1/T_o$, thus $R_l = 1/T_o$. Since the maximal number of parallel transmissions in the spectrum is $\Lambda$ and each regular transmission lasts for $T_{max}$, the maximal returning rate is $\Lambda/T_{max}$. The definition of $T_{max}$ now follows from the fact that in a fully utilized spectrum, we want $\Lambda/T_{max}$ to exceed $1/T_o$. The empirical formula indicates that by increasing $T_{max}$ or reducing the handshake overhead, more parallel transmissions can be supported by the control channel.

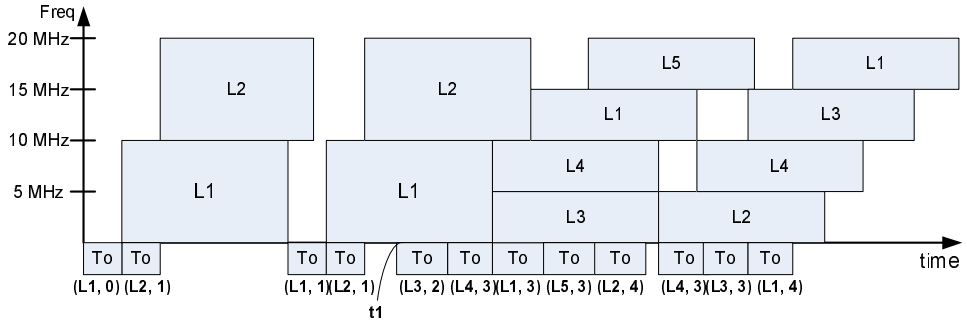


Figure 3.4: Illustration of the spectrum allocation of 20MHz white spaces. Bandwidth options are 10 and 5 MHz. At time $t_1$, Links $L3, L4, L5$ join $L1$ and $L2$. All links are disjoint, blacklogged and within each other's transmission range. The bracket denotes the link ID and the number of valid reserved blocks $N$, at the time of the handshake.

**Estimating $N$:** The algorithm chooses the bandwidth $\Delta f$ for a link $L$ based on $N$, the instantaneous number of disjoint transmissions in the interference range of link $L$. In the distributed scenario, however, it is almost impossible to get a perfect estimation of $N$ considering such a process is repeated in a fine timescale. We therefore approximate $N$ using the number of valid time-spectrum blocks stored in the local table. At time $t_{cur}$, a block is valid if $t + \Delta t > t_{cur}$. Figure 3.4 depicts an example of this online approximation of $N$. Initially, $L1$ and $L2$ each get a half of the spectrum, because the number of pending blocks is 1, hence $N$ is 2 and $\Delta f$ for each link is 10MHz. As three more links join the network, the number of valid blocks increases. Accordingly the protocol forces each link to reduce its bandwidth share $\Delta f$ to 5 MHz in order to increase parallelism. In the initial stage when many new links join the network, it takes a time period to collect relevant reservations and learn the existence of the local contending transmissions. As we show in the simulation section, this learning period is short for various traffic types and even large number of new nodes.

Notice that since $N$ is derived based on the up-to-date reservations, b-SMART is responsive to user and traffic dynamics. The number of disjoint transmissions is effectively tracked especially when flows are long-lived and backlogged with

packets. In the unsaturated case, we find that the method is conservative since $B/N$ constrains the upper limit of the bandwidth. Hence, in such cases, more complex greedy strategies could potentially achieve better performance. Deriving different strategies for deciding the time-spectrum blocks is interesting future work.

**Receiver Scheduling:** To meet the $T_{max}$ duration requirement, our solution incorporates a packet aggregation procedure and implements a round-robin scheduler to handle the packet queues in a node. A node periodically examines the output packet queues for its neighbors. A neighbor becomes *eligible* if (i) it does not have an outstanding reservation, and (ii) the output packet queue for this neighbor has accumulated enough packets to satisfy the $T_{max}$ requirement, or the queue has timed out for packet aggregation.

## 3.4 Performance Analysis

Here we establish a theoretical model that describes the throughput achieved by the KNOWS design in fully connected topologies. We also validate the choice of $T_{max}$ in (3.7) and motivate our selection of control channel bandwidth.

We first analyze the throughput achieved by KNOWS. In line with the rich literature on Markov-based performance modeling and analysis of randomized back-off protocols initiated by Bianchi [34], we focus on the *saturation throughput*. We assume $N$ disjoint flows in a single-hop network in which each sender has backlogged queues with packets of equal size. The minimum bandwidth option is $b_{min}$ and that the white space of bandwidth $B$ can accommodate at most $\Lambda$ parallel transmissions. To show that even for large number of stations ($N > \Lambda$), the control channel can be prevented from becoming a bottleneck by setting $T_{max}$ according to inequality (3.7), we model the control channel using a Markov

model, and verify the model using simulations.

At any given time, let $Q$ denote the number of stations that contend on the control channel. Conversely, the number of stations that are currently transmitting in the white spaces is $N - Q$. In order to make the Markov system analyzable, consider the following simplification. In the saturated case, every time-spectrum block used by a station is of duration $\Delta t = T_{max}$. Because nodes obtain such a block only after a successful handshake on the control channel, the time between two consecutive nodes return to the control channel is at least $T_o$, the average time overhead spent on one successful handshake. Instead of taking into account these complex timing-dependencies between different nodes sending in white spaces, we assume that in any (small enough) time-interval of duration $t$, the probability that a transmitting node returns to the control channel is $\lambda_{ret}(Q) = (N - Q) \cdot t/T_{max}$. On the other hand, the probability that in a time-interval $t$, there is at least one successful handshake on the control channel can be approximated as

$$\mu_{suc}(Q) = \frac{p_{suc}(Q) \cdot t}{p_{suc}(Q)t_{suc} + p_{col}(Q)t_{col} + p_{idle}(Q)t_{idle}}.$$

In this formula, the probabilities $p_{suc}(Q), p_{col}(Q), p_{idle}(Q)$ denote the respective likelihood of a successful handshake on the control channel, a collision, or an idle time-slot, given that $Q$ nodes currently contend on the control channel. Further, $t_{suc}, t_{col}, t_{idle}$ denote the respective duration of such a time-slot. Specifically, it follows from our discussion in Section 3.2 that in b-SMART , $t_{idle}$ equals the empty slot-time $\sigma$ and

$$\begin{aligned} t_{col} &= t_{RTS} + t_{DIFS} + \sigma \\ t_{suc} &= t_{col} + t_{CTS} + t_{DTS} + 2t_{SIFS} + 2\sigma. \end{aligned}$$

We now interpret the probabilities $\lambda_{ret}(Q)$ and $\mu_{suc}(Q)$ as transition probabilities in a one-dimensional birth-death Markov process $\{s(t)\}$ with $\Lambda + 1$ states $C_{N-\Lambda}, \ldots, C_N$. In this process, state $C_i$ signifies a state in which $i$ nodes are contending on the control channel and consequently, $\Lambda - i$ nodes are currently in the midst of a transmission. Let $\mathbf{q} = (q_{N-\Lambda}, \ldots, q_N)$ be the stationary distribution of the chain. It is easy to show by induction that for all $N - \Lambda + 1 \leq i \leq N$,

$$q_i = q_{N-\Lambda} \cdot \frac{\prod_{j=N-\Lambda}^{N-\Lambda+i-1} \lambda_{ret}(j)}{\prod_{j=N-\Lambda+1}^{N-\Lambda+i} \mu_{suc}(j)}.$$

As the $t$ cancel out in the above formulas and because of the additional condition $\sum_{i=N-\Lambda}^{N} q_i = 1$, the only remaining unknowns are the relative success, collision, and idle probabilities $p_{suc}(j), p_{col}(j), p_{idle}(j)$ for all values of $N - \Lambda \leq j \leq N$. For each such $j$, we approximate these probabilities in the steady state using the respective probabilities in a regular 802.11 DCF system consisting of $j$ nodes as derived by Bianchi in [34]. In particular, if $W$ and $m$ denote the minimal size of the contention window and the maximum number of backoff stages, respectively, then

$$p_{suc}(j) = \frac{j\tau(1-\tau)^{j-1}}{1-\theta(j)} \quad \text{and} \quad p_{col}(j) = 1 - \theta(j)(1 - p_{suc}(j)),$$

where $\theta(j) = (1-\tau)^j$ and $\tau$ follows from the unique solution to the nonlinear system of equations in the two unknowns $\tau$ and $p$ defined by $\tau = \frac{2(1-2p)}{(1-2p)(W+1)+pW(1-(2p)^m)}$ and $p = 1-(1-\tau)^{j-1}$. Finally, every time slot is either a success, a collision, or idle and hence, $p_{idle}(j) = 1 - p_{suc}(j) - p_{col}(j)$. This completely specifies the Markov process.

Ideally, the process should always be in state $C_{N-\Lambda}$ or at least remain in states $C_i$ with low $i$ for most of the time, because the throughput achieved in state $C_i$ is proportional to $(N - i) \cdot b_{min}$. Taking this into account, the saturated

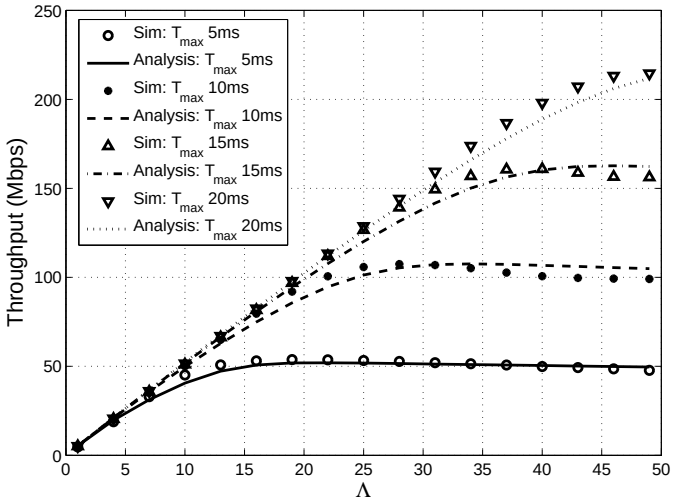


Figure 3.5: Network throughput v.s $\Lambda$

throughput $H$ of KNOWS can be described by the expression

$$H = b_{min} \cdot \frac{T_{packet}}{T_{packet} + T_{ACK}} \cdot \sum_{j=N-\Lambda}^{N} (N - j) \cdot q_j.$$

Figure 3.5 plots the saturation throughput predicted by this model and compares it with our simulation results. The figure shows that the model closely matches our empirical findings.

Our model allows us to study the impact of $T_{max}$ and control channel bandwidth on network throughput. Figure 3.5 shows the throughput performance as $\Lambda$ increases with different $T_{max}$ settings. In particular, the plot shows a setting with $N = \Lambda + 1$; and other parameters are listed in Section 3.5.

As pointed out, our analytical result matches well with the simulations. The reason why the choice of $T_{max}$ is crucial is because it determines the number of parallel transmissions $\Lambda$, in line with the bandwidth of white spaces $B$. When $T_{max}$ is sufficiently large, b-SMART's throughput increases proportionally with $B$. In this specific example, $T_{max} = 20ms$ can support up to 50 parallel transmissions or 250 MHz white spaces. On the other hand, with smaller settings of $T_{max}$,

the network throughput stops from increasing as $\Lambda$ grows over a certain point. At that specific point, the value of $T_{max}$ is the minimal setting of $T_{max}$ to support the corresponding number of $\Lambda$. Further extensive simulation results give strong validation of this relationship between $T_{max}$ and $\Lambda$ and, consequently, the definition of formula (3.7).

The choice of 5 MHz control channel bandwidth tries to balance the following two contradicting aims. The first is to minimize the spectrum resource consumed by the control channel. The other goal is to minimize the average handshake overhead, $T_o$, to control $T_{max}$ within the fine timescale, say tenths of a millisecond. In this sense, the control channel bandwidth determines $T_o$ and hence, contributes to the length of $T_{max}$. Our choice of 5MHz control channel bandwidth to regulate the white spaces in the TV bands has been guided by thorough analysis and simulation study.

## 3.5 Simulation Study

We have implemented the KNOWS design in QualNet [81] and evaluate it extensively in this section. We study two main performance metrics: network throughput and time-spectrum block fairness. We present the results in three phases. First, we microbenchmark the throughput performance of KNOWS, and compare it against SSCH [76], which is a recently proposed multichannel MAC. Second, we show the need of adaptive spectrum allocation by comparing b-SMART against fixed spectrum allocation schemes. Our results show that b-SMART significantly outperforms any single fixed allocation scheme. We then evaluate the impact of spectrum fragmentation on KNOWS, and show that our scheme is able to cope with fragmented white spaces. Finally, we stress test KNOWS under varying degrees of traffic density, packet size and mobility. Our

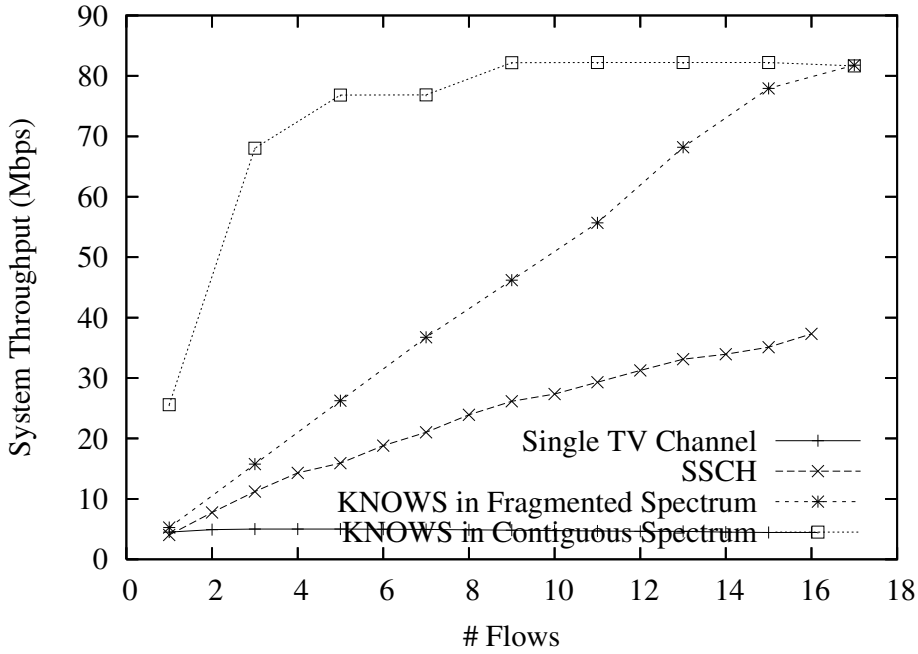


Figure 3.6: System throughput with disjoint UDP flows

results confirm that KNOWS performs reasonably under most scenarios.

In all our simulations, we set the bandwidth of all available white spaces to be 80 MHz. We restrict the bandwidth usable by cognitive radios to be 5, 10, 20 and 40 MHz [3]. $T_{max}$ is set to 5 $ms$ and the control channel bandwidth is 5 MHz. We assume that every 1 MHz of spectrum delivers 1.2 Mbps [49]. Without loss of generality, we adopt the interframe spaces and the physical-layer overhead specified in 802.11a [42]. Unless otherwise noted, all flows are disjoint and backlogged and all nodes are within transmission range of each other. Furthermore, we use a packet size is 1500 bytes unless specified otherwise.

### 3.5.1 Benchmarking Throughput Improvements

We first quantify the throughput achieved by KNOWS. We place all nodes in communication range of each other, and set the flows to be always backlogged. The total vacant spectrum is set to 80 MHz wide, which is approximately half of

[3]These values are supported by the Atheros chipset in our prototype

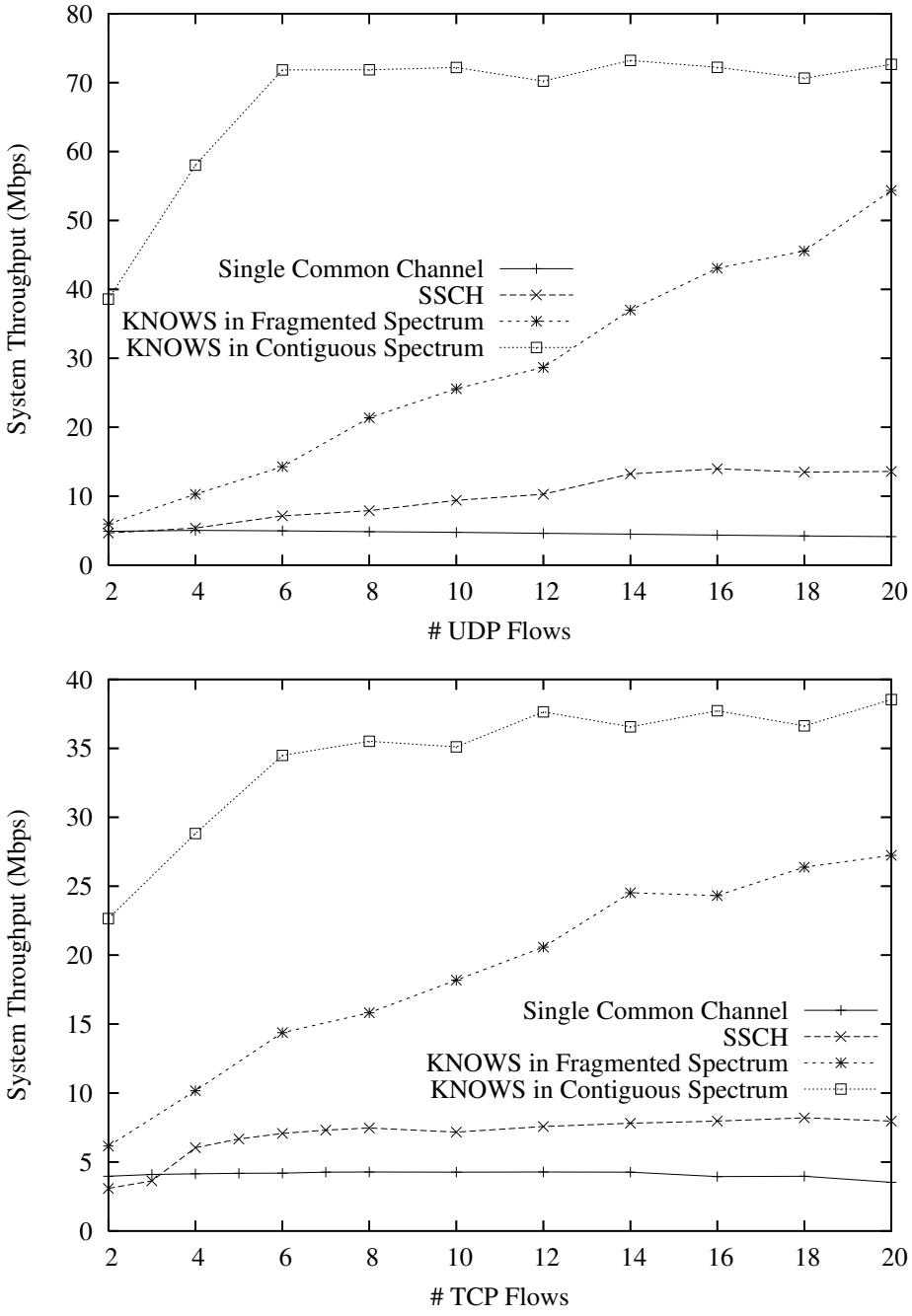


Figure 3.7: Network throughput performance with non-Disjoint flows

the entire UHF spectrum. We run KNOWS in both fragmented and contiguous spectrum. In the first case, the spectrum is fragmented by incumbent TV signals and all the vacant bands are one TV channel wide, i.e. 6 MHz. The contiguous case offers 80 MHz spectrum without any overlapping incumbent operation. For comparison, we simulate SSCH, which is designed to utilize multiple pre-defined channels with *one* transceiver.[4] SSCH assigns a pseudo random seed to each node. The node switches across channels based on the hopping sequence generated by the random seeds. By synchronizing one or more random seeds, SSCH ensures two nodes to meet on certain channel(s) and exchange packets. In this case, nodes in SSCH hop across the vacant TV channels. As a reference, we also run 802.11 MAC in one common TV channel.

#### 3.5.1.1 Disjoint UDP Flows

We first study the throughput as the number disjoint UDP flows increases from 1 to 16. Figure 3.6 shows the system throughput. KNOWS utilizes all vacant TV channels in the fragmented spectrum when increasing the number of flows in the system. Within contiguous spectrum, nodes in KNOWS adjust their bandwidth based on the experienced contention intensity. For example, when there is only one flow in the system, KNOWS assigns it the maximum bandwidth offered by the cognitive radio, i.e. 40 MHz. As we increase the number of flows in the network, each flow uses a smaller bandwidth.

As shown in the figure, KNOWS achieves much higher throughput than SSCH. There are two primary reasons for the increased throughput. Firstly, the adaptive spectrum allocation enables nodes to tune the bandwidth based on the number of disjoint flows. On the other hand, any MAC design using fixed channels cannot adjust the bandwidth to opportunistically exploit the contiguous spec-

[4]We do not know of any MAC protocol that is designed for our radio model of one transceiver and one receiver.

trum. Secondly, KNOWS leverages the extra receiver to perform more optimal spectrum scheduling than the randomized scheduling used by SSCH.

#### 3.5.1.2 Non-Disjoint Flows

We now benchmark the throughput on increasing the number of non-disjoint flows, i.e. flows that may share the same sender or receiver. Figure 3.7 shows the aggregate throughput of UDP and TCP traffic, as the number of flows increases from 2 to 20. The source and destination is chosen randomly for each flow.

In case of contiguous spectrum, the system throughput quickly reaches the maximum throughput. This is because the adaptive spectrum allocations effectively manages the bandwidth across contending nodes. When the spectrum is fragmented, KNOWS takes more time to fully utilize the available bandwidth. This can be explained by senders or receivers shared by different flows, which in turn reduces the possible number of parallel transmissions. However, note that KNOWS still outperforms SSCH because of the reasons described earlier.

### 3.5.2 Effectiveness of Adaptive Bandwidth

We now evaluate the effectiveness of our adaptive bandwidth allocation scheme, b-SMART, in both single-hop and multi-hop mesh networks. We compare against 4 fixed allocation schemes, which assign flows a pre-defined bandwidth block of 5, 10, 20 and 40 MHz respectively. We vary the number of flows in the network and plot the total throughput in the system (a sum of individual flows) in Figure 3.8. In the case of a mesh network, we randomly place 100 nodes in a 500 m x 500 m area, and initiate flows between randomly selected source-destination pairs. We use AODV [79] to discover routes, and the average hop count is 2.5. Figure 3.9 shows the network throughput gain of b-SMSART compared with the fixed allocations in the mesh network.

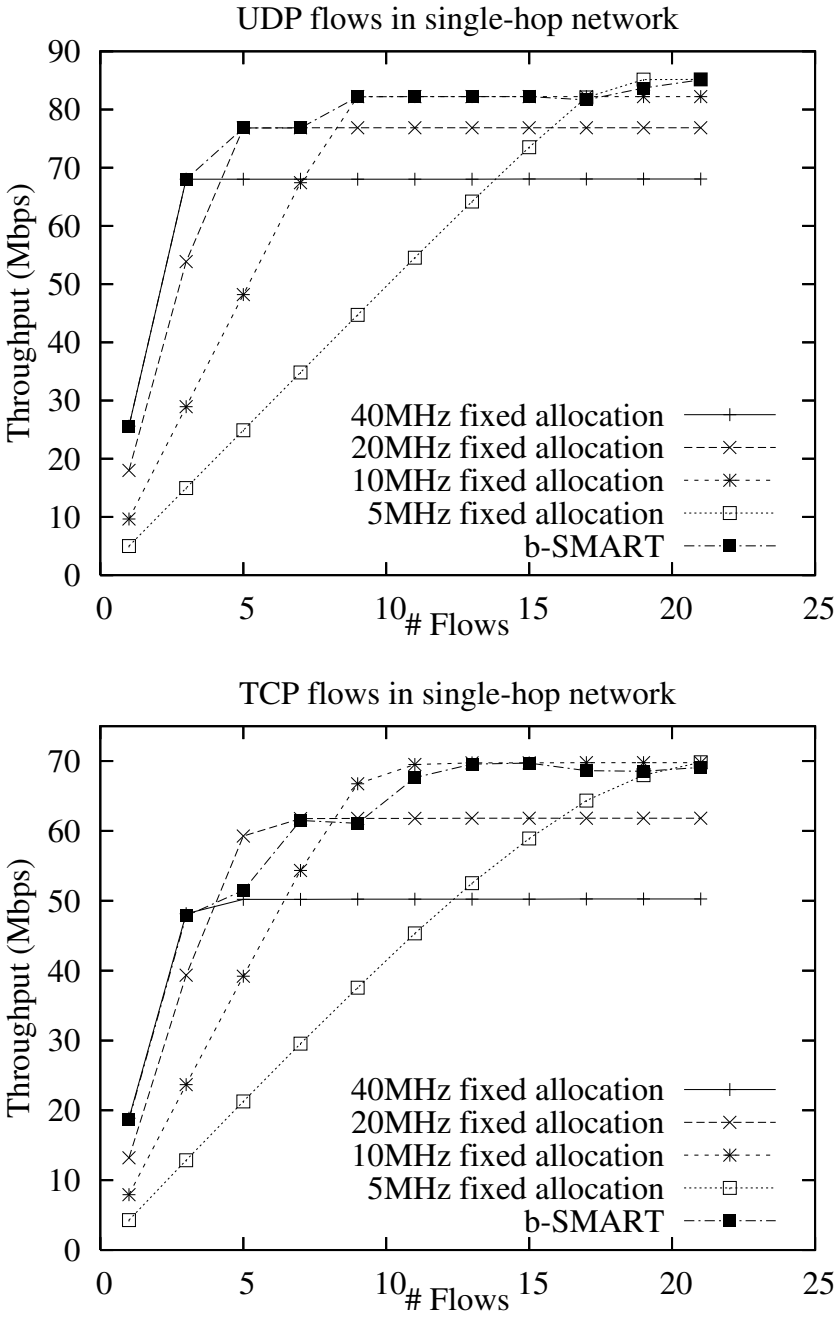


Figure 3.8: b-SMART vs. fixed allocations in mesh networks in single-hop networks

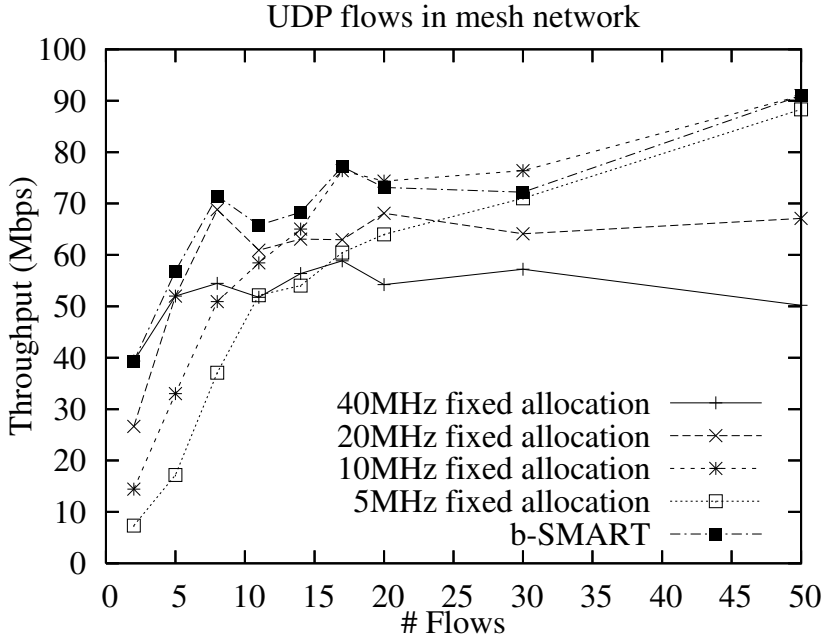


Figure 3.9: b-SMART vs. fixed allocations in mesh networks

As shown in Figure 3.8 and Figure 3.9, no single fixed allocation scheme works best under all scenarios. When there are few flows in the network, fixed allocations with larger bandwidth chunks perform much better since they offer better spectrum utilization. However, when there are more flows in the network, a fixed allocation scheme with smaller bandwidth chunks performs better. In fact, a 5 MHz fixed allocation scheme achieves higher network throughput than a 40 MHz scheme when there are more than 16 flows in the network. This can be explained by the lower data rate used by a smaller bandwidth scheme, which reduces the ratio of per-packet signalling and control overhead (SIFS, DIFS, ACKs) due to longer data transmissions. In contrast, the adaptive bandwidth allocation scheme of b-SMART adapts to use the best among possible bandwidth allocations.

Note that with TCP flows, the proposed spectrum allocation algorithm is slightly worse than the best static allocation in some cases. The reason arises from short ACK packets used by TCP, which are sometimes not big enough to

occupy the minimum-sized time spectrum block. However, in most cases, this does not lead to significant throughput degradation. We also evaluate the time taken by nodes to learn of new TCP and UDP flows and accordingly adapt their bandwidth. In our single hop experiment, we measured the time for 40 nodes to adapt to 20 new flows to be 10 $ms$ for UDP and around $90$ ms for TCP traffic.

### 3.5.3 Impact of Fragmentation

We now study the impact of fragmentation in white spaces in simplified cases, where each spectrum fragment has identical bandwidth. Pattern 1, 2, 3, and 4 refers to fragment sizes of 5, 10, 20 and 40 MHz. We evaluate the throughput performance in single-hop networks for increasing number of *non-disjoint* UDP or TCP flows. In chain networks, where a UDP flow originates at the first node and packets are forwarded to the last node, we vary the number of hops.

Figure 3.10 and Figure 3.11 shows the network throughput achieved by b-SMART in different fragment patterns. In single-hop networks, in every pattern, the throughput increases linearly as the number of flows increase, since b-SMART arranges more parallel transmissions in the fragmented segments. As the fragment size increases, the significant throughput gain is obtained especially when the flow population is less than 12 in this specific case. b-SMART extracts such gain by adaptively bundling the white space when the contiguous spectrum exists. Moreover, the patten with 5 MHz spectrum fragment obtains the lowest throughput because flows are non-disjoint (two flows may share either endpoint) and there is no enough parallel transmission to occupy all 5 MHz fragments. The throughput obtained in the chain network shows a similar trend: b-SMART adaptively adjusts bandwidth for achieving higher performance or for handling fragmentation. In addition, b-SMART achieves a stable throughput after first 2 hops, since our design enables any two non-consecutive hops to use non-overlapped

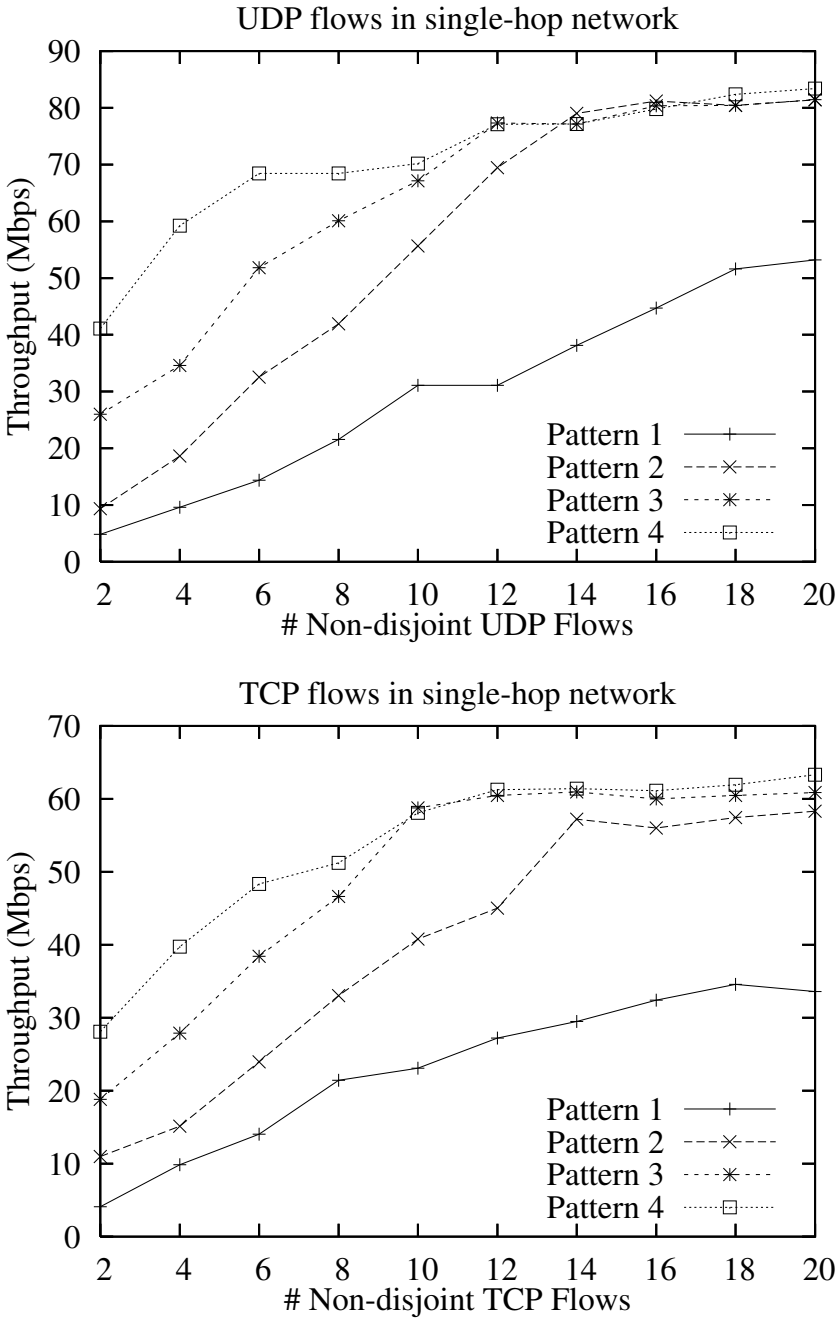


Figure 3.10: Network throughput achieved in different fragmentation patterns in single-hop networks

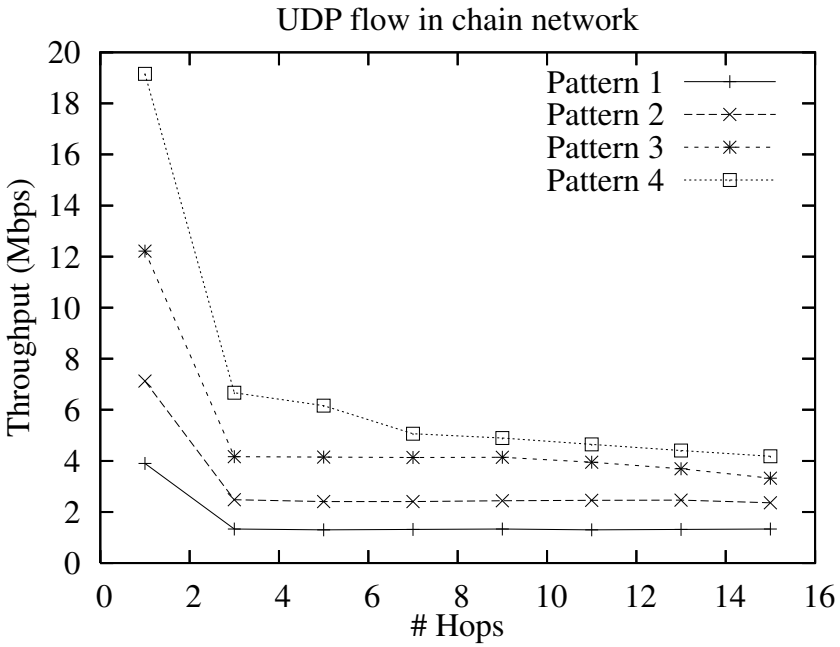


Figure 3.11: Network throughput achieved in different fragmentation patterns in chain network

blocks and transmit in parallel.

### 3.5.4 Impact of Traffic Density and Mobility

Figure 3.12 illustrates the network throughput as the load per flow varies from 100 Kbps to 20 Mbps. We repeat the experiment for the flow populations of 6 and 20 in different fragmentation patterns. In case of 6 flows, when the traffic density is low, the network throughput is identical among different patterns. As the per-flow load increases over 5 Mbps, b-SMART achieves much higher throughput performance in the pattern with large fragment sizes by opportunistically bundling spectrum. With 20 flows, b-SMART obtains similar network throughput by adaptively adjusting bandwidth to enable as many parallel transmissions in white spaces. We also evaluate the fairness in terms of time-spectrum block using Jain's fairness index [27]. The block fairness corresponds to throughput fairness since the transmission rates used by nodes are equal. The

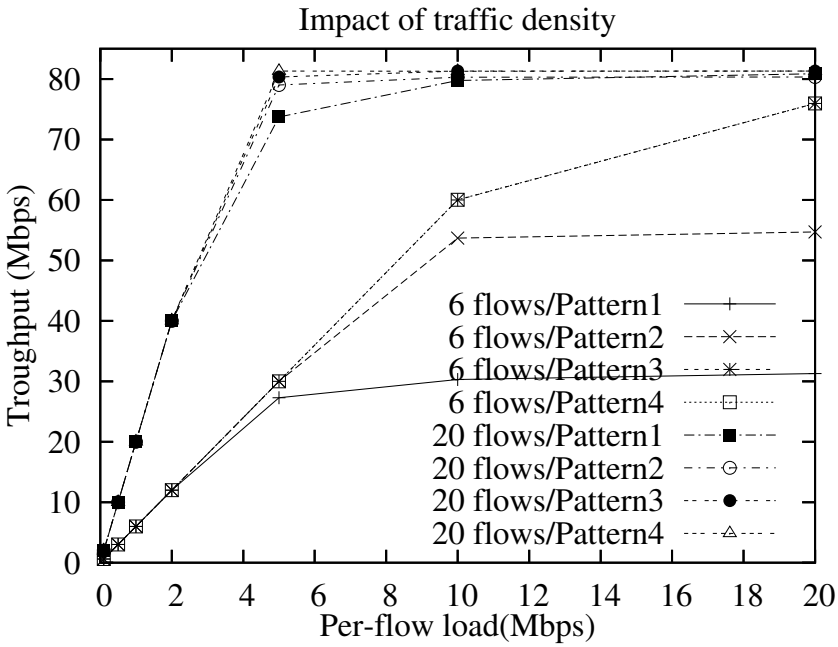


Figure 3.12: Network throughput vs. traffic load

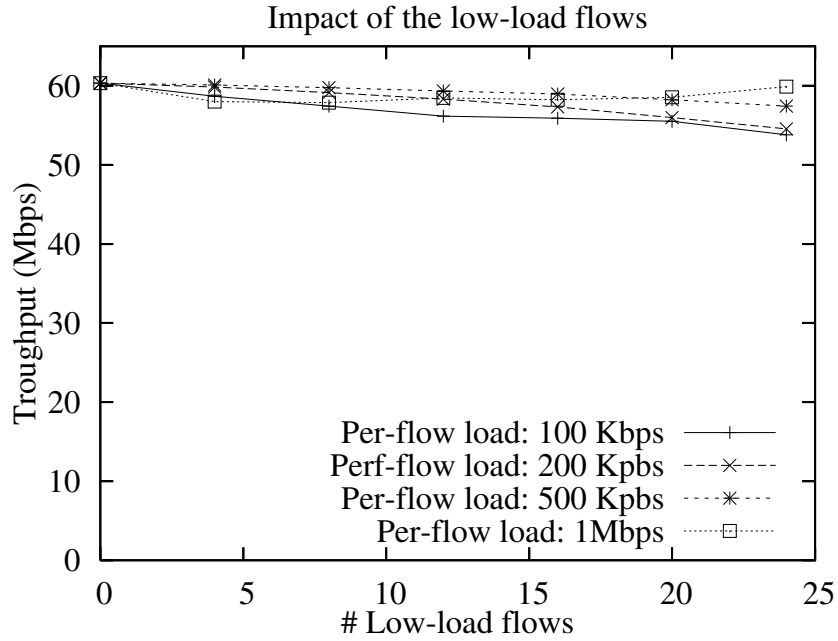


Figure 3.13: Network throughput vs. low-load traffic

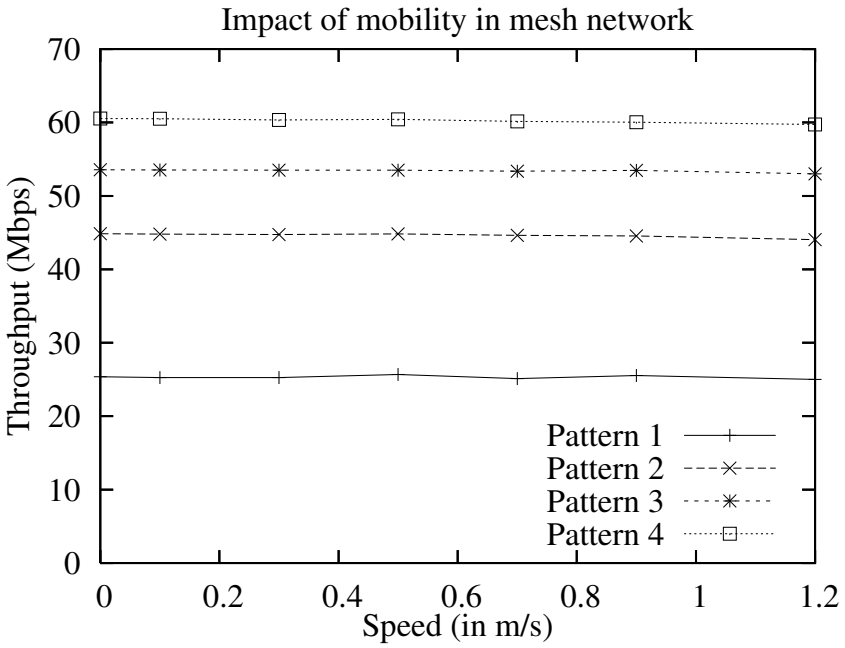


Figure 3.14: Network throughput vs. mobility

Jain's index in every case is close to 1, which validates the block fairness provided by b-SMART.

We next quantify the impact of low-load traffic on the performance of b-SMART in a contiguous 80 MHz white space. We start two backlogged flows together with a number of low-load flows. To stress the design, we reduce the packet size of low-load flows to 512 bytes. Figure 3.13 shows that the network throughput decreases (up to 12%) as the number of low-load flows increases. This is because the control overhead for the low-load flows is not sufficiently amortized by the ensuing data transmission.

Finally, we study the impact of the mobility. We use the same multi-hop network as in Section 3.5.2, but allow nodes to move using the Random Waypoint Model. Each node selects a random point, and move towards it with a speed chosen randomly from an interval,($V_{min}$; $V_{max}$]. Upon reaching its destination, the node moves to a new destination after it pauses for a random period between 0 and 10 seconds. We set $V_{min}$ at 0.01 m/s and vary the $V_{max}$ from 0.2 to 1.2

m/s. Figure 3.14 shows the overall throughput using 8 flows when run for over 5 minutes. The throughput experiences minor degradation because b-SMART allocates spectrum at a fine-timescale.

## 3.6 Discussion

We discuss some design choices made for KNOWS and some directions for future work.

In this paper, we propose a new scheme for adaptive spectrum allocation. This scheme is different from the conventional method of spectrum allocation, which divides the available spectrum into fixed channels of equal bandwidth. For example, in IEEE 802.11a, there are 13 orthogonal channels of 20 MHz bandwidth. This fixed channelization structure is simple and incurs low implementation cost. However, such a structure creates hard boundaries for utilizing the entire spectrum. One implication is that it prevents users from bundling vacant channels to obtain higher data rates. Moreover, in the TV spectrum, the spectrum is fragmented by the incumbent signals, leaving various sizes of spectrum segments available for sharing. The adaptive spectrum allocation adopted by KNOWS deviates from this channel concept. The operating frequency and the bandwidth is adaptively determined based on local information.

We use a narrow-band control channel for disseminating spectrum usage information. In contrast to systems that use a central spectrum controller with global knowledge of user activities and spectrum allocations, KNOWS uses a distributed approach for efficient spectrum sharing. Each node constantly listens on the control channel to keep track of spectrum availability in real time. In recent work [56], we have explored the tradeoffs involved in separating control traffic from data, and use the results to set the control channel bandwidth to be 5 MHz.

Our current design uses a 5 MHz band in the unlicensed ISM spectrum (902–928 MHz) as the control channel.

We note that using one fixed control channel raises security concerns. The nodes in KNOWS cannot operate in the TV spectrum if the control channel is occupied or jammed. To improve the robustness, we are investigating the use of a common hopping sequence to build the control channel. The control channel can hop across the vacant TV channels according to a negotiated sequence at a coarse-time level (several seconds). Hence, the single point of failure caused by using a single control channel can be largely reduced. In addition, the control channel is different from the frequency band used for data communications. Different bands may have different propagation proprieties, especially in terms of the transmission range. We are conducting experimental studies using our prototype radios to quantify the effect of transmission range mismatch. We expect our results to be consistent with prior work [56].

## 3.7 Related Work

We summarize and compare prior work relevant to KNOWS mainly from the spectrum sharing perspective, which defines how the vacant spectrum should be shared among unlicensed users.

There are two different approaches for supporting spectrum sharing: centralized control and distributed coordination. In the centralized control category, IEEE 802.22 [41] is the first standardization effort to define unlicensed operations in the TV spectrum. In 802.22, a base station serves multiple Consumer Premise Equipments (CPEs) and determines the availability of a TV channel by combining scanning results from the CPEs. The base stations are allowed to combine three contiguous TV channels to generate an 18 MHz-wide operating band. Two

other centralized systems are DIMSUMnet [62] and DSAP [92]. In DIMSUMnet, the spectrum brokers coordinate spectrum usage in relatively large geographic region; in DSAP [92], the centralized controller manages the spectrum access by offering long-term leases to secondary users. In contrast to the above systems, KNOWS is based on distributed coordination, and is not lease based.

Within the distributed category, several MAC protocols have been proposed to utilize the overall spectrum. However, to the best of our knowledge, all of them are based on static, evenly-divided channels. For example, SSCH [76], MMAC [86], and LCM-MAC [64] use a single radio to exploit multiple fixed channels. DCA [94], xRDT [64], HMCP [78] are proposed to use multiple channels in parallel with multiple radios. The existing MAC solutions assume a fixed channel as the default spectrum allocation unit. However, channels are not well defined in the TV bands due to the dynamic nature of white spaces. Should the bandwidth be the size of a TV channel or should it be smaller or larger? Where should we set the center frequency? These questions have motivated KNOWS to reconsider the essence of spectrum allocation. In our system, nodes adaptively utilize different frequencies and bandwidth based on spectrum availability and contention in the network.

Several MAC proposals have addressed different issues in cognitive radio networks. HD-MAC [103] maintains connectivity in a large network using a set of control channels, where each control channel manages a different local group. This is in contrast to using a global control channel. Coordination between local groups merges different groups into a connected network. In contrast, KNOWS uses a narrow channel in the unlicensed band as the common control channel. DC-MAC [104] conducts a theoretical study to derive decentralized strategies for unlicensed users to sense and access fixed channels. DOSS [63] allows nodes to use a variable bandwidth channel based only on the spectrum availability. In

comparison, KNOWS provides a detailed algorithm to adapt the bandwidth at a fine-time-scale and decides the time duration considering the traffic load information.

## 3.8 Chapter 3 In Brief

The advent of advanced radios and the prospect of abundant free spectrum has spurred a flurry of research activities in cognitive radio networks. To address the challenges posed by the dynamic nature of white spaces, we have presented KNOWS, which is a system encompassing new hardware, an enhanced MAC protocol and spectrum sensing capabilities, for efficiently utilizing unused portions of the licensed spectrum. KNOWS cooperatively detects incumbent operators and efficiently shares the vacant spectrum among unlicensed users. Our hardware consists of a development board with a scanner/receiver radio and a reconfigurable transceiver. We propose a new MAC design called CMAC, which enables a spectrum reservation scheme in addition to the virtual sensing approach of IEEE 802.11.

We then introduce the time-spectrum blocks as the units of spectrum reservation, and use them to formalize the spectrum allocation problem over white spaces. We show that this problem is radically different from any existing wireless systems based on the fixed channelization. We also propose and evaluate a centralized algorithm and a distributed solution called b-SMART, which enables nodes to share the white spaces in a fine timescale. b-SMART maintains up to date information about spectrum usages of all its neighbors, and stores it in a local table. And then b-SMART uses the collected information to dynamically decide on the portion of the spectrum assigned to a given transmission to maximize parallelism and reduce interferences. Using the analysis and simulation

studies, we have shown that KNOWS significantly increases network capacity as compared to IEEE 802.11 based systems.

# Chapter 4

# Improving WLAN via Cognitive Radios

*That which we persist in doing becomes easier,*
*not that the task itself has become easier,*
*but that our ability to perform it has improved.*
*— Unknown*
*To change and to improve are two different things.*
*— German Proverb*

IEEE 802.11-based WLANs use a simple, fixed channelization structure, which divides the available spectrum into a set of channels of equal bandwidth. The fixed channelization has severely constrained the total capacity and fairness of WLANs. More specifically, recent measurements of WLANs in real world [18, 23, 25] show significant spatial and temporal variations in user and traffic load. User populations served by different access points (APs) in a WLAN fluctuate considerably over time, and are extremely unbalanced. Certain APs become hotspots while others remain unused. The fixed channel structure offers very limited flexibility in handling such variations.

In this chapter, we first propose a radically new channelization structure in Section 4.1. Our scheme dynamically creates the appropriate number of channels to accommodate all interfering APs, and adjusts the channel bandwidth (channel-width) depending on the instantaneous traffic loads at the AP and its the neighboring APs. The dynamic channelization is enabled by recent advances in radio technologies, such as cognitive radios, which make it possible to reconfigure the key operating parameters of radios, including center-frequency, bandwidth, and power, with a very low time overhead [98]. In Section 4.2, we propose a scalable MAC design to cope with variations of clients who operate in the same channel and compete for channel access. Results from analysis and extensive simulations demonstrate that our proposed channelization structure and the scalable MAC design effectively handle variations of user and traffic load. The proposed schemes of dynamically allocating variable-width channels and scalable MAC work together to improve network capability and fairness in WLANs.

## 4.1 Adaptive Channel-width Assignment

We envision a network architecture in which the bandwidth of different APs can be adapted according to their respective traffic intensity. Hotspots with many clients will get wider channels at the cost of neighboring APs with little load, which will receive less bandwidth. We begin the section with an overview of our assumptions regarding our system and architecture. Based on this, we formulate a theoretical model which allows us to go on and formulate our algorithms in Sections 4.1.3 through 4.1.5.

### 4.1.1 System Assumptions

We primarily consider enterprise networks in which all APs are connected via a backbone network. Each access point is capable of obtaining some measure that represents its current *load*. A simple measure could be the number of clients currently associated with this AP, but more sophisticated and accurate measures that take into account the traffic demands of each client may be preferable. At any rate, each AP periodically reports its load to a centralized server that is attached to the network's backbone network and maintains a view of the traffic distribution across the network in a local database.[1] Periodically, the centralized server—based on information stored in its database—runs an algorithm that computes an optimal or near-optimal allocation of channel-widths and center-frequencies to APs. Once computed, it sends the allocations to the respective APs which, along with their associated clients, switch to the new center-frequency and channel-width.

Besides the flexibility to assign more bandwidth to certain APs, bandwidth allocation must also be adaptive in a temporal sense. That is, in order to react to mobility and the dynamic nature of user demand at different APs, bandwidth allocation should not be static in time, as it is in the standard IEEE 802.11 architecture. The centralized server therefore reassigns new bandwidths and center-frequencies to APs periodically, say in intervals of 10 minutes. Alternatively, the spectrum allocation may be updated whenever a *threshold of suboptimality* is surpassed. That is, APs switch to a new bandwidth assignment only when the efficiency of the currently used assignment degrades below a certain point in comparison to the optimal reassignment.

[1] Alternatively, using more decentralized, distributed solutions are also possible and an interesting direction for future research. Since the main focus of this work is to identify and quantify the potential gain when abandoning fixed bandwidth channels in IEEE 802.11, we focus on the conceptually simpler centralized solution.

Efficiently setting up and managing a Wireless LAN network poses challenging and complex problems. Several degrees of freedom may be tuned to optimize the network's throughput and/or fairness, including transmission ranges (cell breathing [17]), data rates, load balancing schemes, modulation schemes, density of deployment, and even the locations of the APs. In the sequel, we assume these variables to be fixed (e.g., we assume that each AP has configured the transmission power to obtain the uniform transmission range in different bandwidth settings.), which allows us to more closely study the impact of flexible and dynamic bandwidth allocation on WLAN efficiency. Doing so keeps our results clean from complex inter-dependencies. On the other hand, it is clear that by simultaneously optimizing over multiple tuning parameters (e.g., by combining our adaptive-bandwidth allocation with cell-breathing), even better results are achievable.

Further assumptions we make is that the achievable data rate is linear to the available bandwidth [22]. Also, we make the conservative assumption that overlapping bandwidths always interfere. That is, we seek to assign non-overlapping frequency interval to any two interfering access points.

Clearly, numerous problems of utmost practical importance remain. For instance, since bandwidths of different APs are variable and dynamic in time, there needs to be an efficient method for clients to discover the bandwidth and center frequency of its neighboring access points. Also, the process of an AP (along with its associated clients) switching to a different center-frequency and bandwidth must be smooth and seamless. A more detailed discussion of these and other important practical issues (including the issue of legacy clients) follows in Section 4.1.7.

### 4.1.2 Problem Formulation and Notation

The main algorithmic problem involved in the system architecture sketched in the previous section is the selection of appropriate channel-widths and center-frequencies. We study a simple network model that allows us to characterize the potential gain of our novel channelization approach. It also allows us to analyze and understand the respective merits of different allocation algorithms. The model makes several simplifying assumptions, but manages to capture those characteristics that govern the design of appropriate algorithms for our bandwidth selection problem. As we focus on the impact of having different channel-widths at APs, we assume each AP to have a fixed (but not necessarily uniform) transmission power $P_i$ and fixed location.

Let the network consist of $n$ access points $AP_1, \ldots, AP_n$. Given the fixed locations and transmission powers, we can determine a *conflict-graph* $G = (V, E)$ of the wireless network as follows [48, 53]: Every AP is represented by a node $i \in V$ and there is an edge between two APs if they have significantly overlapping coverage regions and should therefore avoid transmitting on the same frequency. Practically, we model an edge $(i, j) \in E$ if simultaneous transmission of both $AP_i$ and $AP_j$ could result in harmful interference at some client in the network. Clearly, this binary model of interference is a tremendous simplification of physical reality [71]. In the context of our work, however, it is justified as it is conservative and ignores additional optimizations that could further enhance our system.

In our practical system, the interference relationship between neighboring access points can be determined in an ad hoc fashion (e.g., by APs using beacon messages to probe their proximity to other APs, or by client feedback) as proposed for instance in [48]; or it may be statically provided as part of the network planning. In any case, the conflict graph is static and needs to be updated only

rarely, therefore posing no serious practical problem on our system design. For an $AP_i$, we denote by $N(i)$ the set of all neighboring APs that are potentially in conflict with $AP_i$, $N(i) = \{AP_j \mid (i,j) \in E\}$.

Let the total *demand* of clients that are associated to $AP_i$ be denoted by $D_i$ bit/s. This demand, along with the interference graph, forms the input to the spectrum assignment algorithm running in the centralized server. The *load* that an AP can serve depends on its clients' demand and, crucially, on its *channel-width*. Let $B_i$ be the channel-width allocated to $AP_i$ and let $B_{tot}$ be the total system bandwidth available. As pointed out, it can be modeled as

$$L_i = \min\{\chi B_i, D_i\}, \tag{4.1}$$

where $\chi$ is a constant that captures how efficiently the available frequency spectrum can be utilized [22]. With standard modulation techniques, this constant is roughly $\chi \approx 1$.[2]

**Dynamic-Width Channel-Assignment Problem:** The dynamic-width channel-assignment problem in infrastructure-based wireless networks asks for a *non-interfering assignment* of a start frequency $S_i$ and a bandwidth $B_i$ to each access point $AP_i$. The access point $AP_i$ uses the frequency band $\mathcal{I}_i = [S_i, S_i + B_i]$ for serving its clients and satisfies a load of $L_i = \min\{\chi B_i, D_i\}$. The assignment is *non-interfering* if the assigned intervals $\mathcal{I}_i$ and $\mathcal{I}_j$ of any two neighboring APs $i$ and $j$, $(i,j) \in E$, is non-overlapping.

A practical algorithm for the dynamic-width channel-assignment problem should achieve two goals: *high throughput* and *fairness*. The former is achieved by maximizing system throughput $L_{Sys} = \sum_{i \in V} L_i$. For fairness, various definitions can be considered and the optimization criterion can be defined appropri-

[2]Formula 4.1 abstracts away the fact that different frequency bands have different signal propagation characteristics. Within the spectrum and bandwidth range studied in this paper, however, the formula is a reasonable approximation.

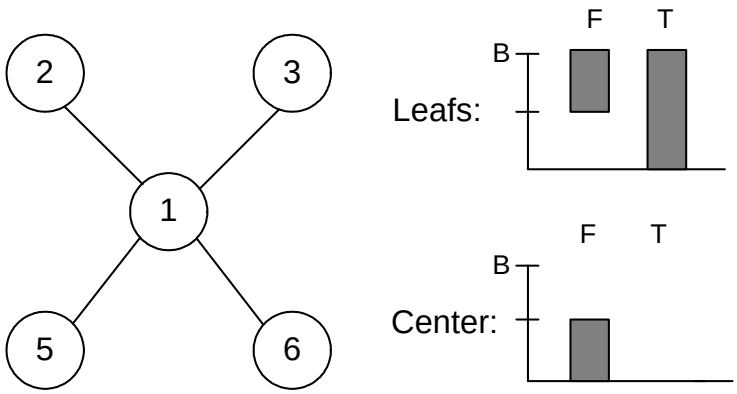


Figure 4.1: Network in which a throughput-optimal solution is unfair. T and F denote the allocations in a throughput-optimal and fair solution, respectively.

ately. The difficulty is that in many cases, achieving high system throughput and fairness are contradicting aims. Consider the star graph with uniform demands shown in Figure 4.1. An allocation maximizing system throughput assigns each leaf AP the entire spectrum, while giving no bandwidth to the center AP. While achieving maximum throughput, such a solution starves clients associated to the AP in the center. A completely fair solution, on the other hand, consists of assigning each AP a channel-width spanning half of the totally available spectrum. In this paper, we address this fairness vs. throughput trade-off by a simple practical solution: We fix a lower bound on the degree of fairness that must be maintained between different APs and strive to optimize the system throughput under this condition.

### 4.1.3 Optimal Solution

The *dynamic-width channel-assignment problem* is fundamentally different from *coloring problems* or *multicoloring problems* that have been extensively studied in the networking community. The reason is that, unlike in (multi)coloring problems, the interval assigned to each AP must consist of a *contiguous* chunk of spectrum of various sizes. This contiguity constraint that does not exist in coloring

problems can lead to *fragmentation* of the spectrum. When spectrum becomes fragmented, APs may be unable to reserve a large contiguous part of the spectrum even though the totality of unused spectrum would be sufficiently high. Besides its practical importance, the problem is thus of great theoretical importance as it combines the complexity of both coloring and packing problems.

It is possible to characterize the optimal solution of a problem instance by means of an integer linear program (ILP). Let $b_i$ and $s_i$ be variables that denote the bandwidth and start frequency allocated to $AP_i$. Further, for each pair of APs $i$ and $j$ with $(i,j) \in E$, we use two binary indicator variables $f_{ij}$ and $f_{ji}$. The following ILP determines the optimal system throughput achievable in a network with arbitrary channel-width options.

$$
\begin{aligned}
\max \quad & \sum_{AP_i \in V} b_i & & \\
s_i + b_i - s_j - f_{ij} \cdot B &< 0 & &, \forall (i,j) \in E \\
s_j + b_j - s_i - f_{ji} \cdot B &< 0 & &, \forall (i,j) \in E \\
f_{ij} + f_{ji} &\leq 1 & &, \forall (i,j) \in E \\
s_i + b_i &\leq F_{top} & &, \forall\ AP_i \in V \\
s_i &\geq F_{bottom} & &, \forall\ AP_i \in V \\
\chi \cdot b_i &\leq D_i & &, \forall\ AP_i \in V \\
f_{ij}, f_{ji} &\in \{0,1\} & &, \forall (i,j) \in E
\end{aligned}
$$

The first two constraints force the auxiliary variables $f_{ij}$ and $f_{ji}$ to behave as follows. The variable $f_{ij}$ is 1 if and only if the top-frequency $s_i + b_i$ of $AP_i$'s spectrum interval is "above" (higher frequency) than the lower end $s_j$ of $AP_j$'s interval. Conversely, $f_{ji} = 1$ if and only if $s_j + b_j > s_i$. Considering two intervals $[s_i, s_i + b_i]$ and $[s_j, s_j + b_j]$, it is easy to observe that these intervals overlap if and

only if $s_i + b_i > s_j$ and $s_j + b_j > s_i$, i.e., if the top-frequency of both intervals are higher than the start-frequencies of the respective other interval. The third constraint therefore guarantees that no two neighboring intervals in the graph overlap, i.e., the resulting channel assignment is non-overlapping. The remaining constraints are straightforward. The first two ensure that the assigned interval is located within the available spectrum $[F_{bottom}, F_{top}]$. And finally, the sixth one expresses that raising the bandwidth above the demand does not increase throughput.

The important aspect missing in this ILP formulation is *fairness*. However, fairness conditions can easily be integrated into our ILP by adding additional constraints. In our evaluation section, for instance, we consider a fairness condition in which every AP is guaranteed to receive at least its fair share of bandwidth in its neighborhood. In particular, we define $\phi(i) = D_i/(D_i + \sum_{j \in N(i)} D_j)$ as the *minimum fair spectrum-share* that $AP_i$ should receive. We can then enforce this notion of fairness by adding the following constraint to the ILP: $b_i \geq \alpha\phi(i) \cdot B_{tot}, \ \forall \ AP_i \in V$. The constant $\alpha$ characterizes the trade-off between fairness and throughput. The smaller $\alpha$, the more flexibility the ILP solver has to sacrifice fairness in order to improve throughput. Other notions of fairness can similarly be included into our ILP formulation.

The ILP formulation assumes start-frequencies and channel-widths to be arbitrarily tunable. This is in contradiction to existing hardware platforms which typically have a small limited number of *bandwidth options*, a set of available channel-widths to which the transceiver can be tuned. Discrete sets of bandwidths can easily be incorporated in our ILP formulation by restricting the variables $b_i$ to belong to a corresponding set of integers. In Section 4.1.8, we examine the impact of this discrete set of bandwidth options.

While the ILP formulation describes the theoretical optimum of any prob-

lem instance, it is computationally practicable only in small networks. Specifically, the dynamic-width channel-assignment problem is NP-hard and hence, unless $P = NP$, there exists no efficient solution for its ILP formulation. For the sake of simplicity, we present a simplified version of the theorem that proves hardness only for $\alpha > 2/3$.

**Theorem 4.1.1.** *The dynamic-width channel-assignment problem problem is NP-hard for any fairness parameter $\alpha > 2/3$. This holds even in restricted geometric graph models such as the unit disk graph.*

*Proof.* The proof is by reduction to the 3-coloring problem of a graph, which is known to be NP-complete even in unit disk graphs [37]. Given an instance $G = (V, E)$ of the 3-coloring problem, construct an instance $G' = (V', E')$ of the dynamic-width channel-assignment problem as follows. For each $v_i \in V$, create 7 APs $AP_i^1, \ldots, AP_i^7$ and connect them to build three triangles as $(AP_i^1, AP_i^2)$, $(AP_i^1, AP_i^3)$, $(AP_i^2, AP_i^4)$, $(AP_i^2, AP_i^5)$, $(AP_i^3, AP_i^6)$, and finally, $(AP_i^3, AP_i^7)$ (cf Figure 4.2). Further, assume that for each $i$, $AP_i^2$ and $AP_i^3$ have $1/(\alpha - 2/3)$ backlogged clients, and all other APs have one client. When scaling, $D_i^2 = D_i^3 = \alpha - 2/3$ and all other demands are 1. Finally, for each $(v_i, v_j) \in E$, add a link $(AP_i^1, AP_j^1)$ to $E'$. Observe that due to the fairness condition, every feasible solution must assign APs $AP_i^2$ and $AP_i^3$ a spectrum block of width at least

$$B_i^2 \;=\; B_i^3 \;\overset{!}{\geq}\; \alpha\phi(i)B_{tot} \;\geq\; \frac{\alpha/(\alpha - 2/3)B_{tot}}{2/(\alpha - 2/3) + 3} = \frac{B_{tot}}{3}.$$

We first show that if $G$ is 3-colorable (yes-instance), the total system throughput is at least $T_{Sys}^{yes} \geq 7|V|B_{tot}/3$. Since $G$ is 3-colorable, the graph induced by the APs $AP_i^1$ can also be colored using three colors. Since each gadget itself can also be colored using three colors (regardless of the specific color assigned to its connector AP $AP_i^1$), it follows that the entire graph $G'$ is 3-colorable. The lower

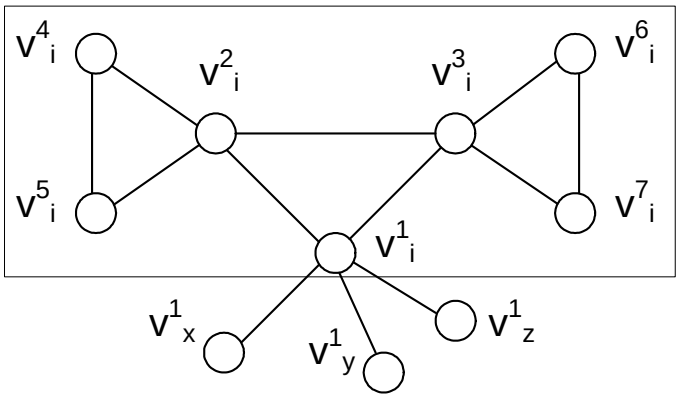


Figure 4.2: The gadget representing a node in $G$.

bound on $T_{Sys}^{yes}$ is now easily obtained by assigning each AP with colors 1, 2, and 3 the spectrum $[F_{bottom}, F_{bottom} + B_{tot}/3]$, $[F_{bottom} + B_{tot}/3, F_{bottom} + 2B_{tot}/3]$, and $[F_{bottom} + 2B_{tot}/3, F_{bottom} + B_{tot}]$, respectively.

Next, we show that if $G$ is not 3-colorable (no-instance), the total system throughput $T_{Sys}^{no}$ is strictly less than $7|V|/3$. Since the subgraph induced by APs $AP_i^1$ is not 3-colorable, there must exist at least one AP, say $AP_x^1$, that is assigned a channel-width of at most $B_{tot}/4$. The total throughput achieve by APs $AP_x^1, \ldots, AP_x^7$ is then at most $(2 + 1/4)B_{tot}$. Also, because all APs $AP_i^2$ and $AP_i^3$ have a bandwidth of at least $B_{tot}/3$, no AP $AP_i^1$ in $G'$ can have a higher bandwidth than $B_{tot}/3$. Hence, the total throughput is at most $|V-1|B_{tot} \cdot 7/3 + (2+1/4)B_{tot} < T_{Sys}^{yes}$. This concludes the proof. □

While the ILP formulation can thus be used to compute optimal assignments in small-scale networks, this approach does not scale. Therefore, we now investigate computationally efficient approximate solutions.

### 4.1.4 LP-Based Approximation

As mentioned earlier, whereas the problem of channel assignment in the conventional channelization framework can be modelled as graph coloring, a key

new flavor in our problem is the need for avoiding *fragmentation*. Specifically, we need to assign one interval to each node, which does not overlap with the intervals assigned to its conflicting nodes (neighbors in the conflict graph). We have degrees of freedom in deciding how long the intervals should be and in deciding where to put them.

#### 4.1.4.1 A Packing Algorithm that avoids Fragmentation

We start by first studying the packing problem in isolation. Assume that the widths of the bandwidth interval allocated to each AP was already determined. How should we efficiently place these intervals? Intuitively, adhering to the following rules of thumb may help:

R1. Pack large items first.

R2. Try to fill up from one end.

Besides being a packing problem, our channel-bandwidth assignment problem also has the flavor of a complex (interval) *coloring* problem. In greedy coloring algorithms, nodes are visited one-by-one, and each node tries to reuse some existing color if possible selecting a new color only if necessary. Clearly, this procedure colors any graph using at most $\Delta(U)+1$ colors, where $\Delta(U)$ is the maximum node-degree. Similarly, if we were not constrained to assign a *contiguous interval* to each AP, we could assure that all required bandwidth can be packed in a total bandwidth of

$$\delta(\boldsymbol{b}) \stackrel{\Delta}{=} \max_{u \in V} \Big( b_u + \sum_{v \in N(u)} b_v \Big), \tag{4.2}$$

which is essentially the continuous counterpart of the $\Delta(U)+1$ upper-bound. That is, without the contiguity constraint, the greedy coloring algorithm assures that the total bandwidth requirement is $\delta(\boldsymbol{b})$.

We now present an approximation algorithm that combines both the packing and coloring aspects of the problem. Assume that the sizes of all bandwidth intervals followed a power series, i.e., each interval has length $2^k$ for some integer $k$. Applying rule of thumb #1, we sort the items in decreasing order of their sizes and try to pack them one by one into the real axis $[0, +\infty]$. Applying rule of thumb #2, when packing each item, we always try to fill up from one end, closer to the origin. When packing in this way, it can be proven by induction that whenever an interval of size $2^k$ is packed, all available intervals (the spectrum gaps still available) are of size at least $2^k$ (in fact, they are an integer multiple of $2^k$). Hence, in this case, we do not suffer from fragmentation and as pointed out before, the total bandwidth required to pack all intervals is at most $\delta(\boldsymbol{b})$. Therefore, this method achieves for the joint packing and coloring problem the same performance that one can achieve for coloring.

If the bandwidth intervals to be packed do not follow a power series, we can round them accordingly. Suppose the given interval lengths are $b_0 \geq b_1 \ldots \geq b_N$. Then we round each $b_i$ to $\tilde{b}_i = \lceil b_i/b_0 \rceil * b_0$, where the $\lceil x \rceil = 2^{-k}$, for some integer $k$. Consequently, all intervals can be packed within a maximum length of

$$\max_{u \in V} \left( \tilde{b}_u + \sum_{v \in N(u)} \tilde{b}_v \right) \leq 2\delta(\boldsymbol{b}). \tag{4.3}$$

Finally, we can linearly map the assigned frequencies in $[0, 2\delta(\boldsymbol{b})]$ to the entire available spectrum interval $[F_{bottom}, F_{top}]$. Doing so, we have packed demands $\boldsymbol{b}$ in a maximum interval of $2\delta(\boldsymbol{b})$, which is at most by a factor of 2 (due to the rounding) worse than applying the greedy coloring algorithm to a relaxed problem where each node can make use of non-contiguous bands.

#### 4.1.4.2 Optimizing the interval lengths

The packing algorithm presented in the previous subsection is effective in assuring the performance for the worst AP (with maximum demand in its neighborhood). While this is good from the fairness perspective, it may harm throughput in scenarios in which some parts of the graph are dense, and others are sparse. (Consider for instance a dense clique and a line-network attached to it. Due to the linear scaling at the end of the packing procedure, APs on the line will not utilize the available spectrum efficiently). In this section we present a method for enhancing the overall throughput without sacrificing fairness. We use the packing algorithm as a building block that packs any demand vector $\boldsymbol{b}$ into an spectrum of width $[0, 2\delta(\boldsymbol{b})]$. The idea is to employ linear programs to search for a demand vector with good worst-case performance $\delta(\boldsymbol{b})$ and good overall throughput. We then run the packing algorithm over the resulting demand vector $\boldsymbol{b}$ to pack it into $[0, 2\delta(\boldsymbol{b})]$.

Consider the following linear program:

$$B_{\text{total}}(\alpha) \triangleq \max_{\boldsymbol{b}} \sum_{u} b_u, \quad \text{subject to:} \tag{4.4}$$

$$b_u \geq \alpha\phi_u \cdot B_{tot}, \quad \forall u \tag{4.5}$$

$$b_u + \sum_{v \in N(u)} b_v \leq B, \quad \forall u. \tag{4.6}$$

Constraint (4.6) ensures that the computed vector $\boldsymbol{b}$ results in a feasible solution with a greedy coloring algorithm. Constraint (4.5) maintains fairness by guaranteeing node $u$ a resource share of $\alpha b_u$. By varying the constant scaling parameter $\alpha$ from 0 to some maximum value $\alpha^*$, different tradeoffs between fairness and throughput efficiency can be achieved. Using the maximum value $\alpha^*$ maximizes the worst node's performance; this value can be determined using the following

LP:

$$\alpha^* = \max_{\boldsymbol{b},\alpha} \alpha, \quad \text{subject to:} \quad (4.5)(4.6) \tag{4.7}$$

**Practical Deployment:** Our LP-based algorithm leaves open various parameters for tuning the involved fairness vs. throughput trade-off. A simple way of employing it in practice is the following: First, determine the optimal fairness parameter $\alpha = \alpha^*$ using LP 4.7. Then, using this $\alpha$, use the first LP to compute $B_{\text{total}}(\alpha)$. This amounts to a conservative approach that maximizes the sum throughput (by "flattening" the demands at the nodes) while assuring the maximum level of fairness at the worst node. The LPs can either be solved directly using an LP solver, or we can apply efficient approximation algorithms for so-called packing LPs [35].

#### 4.1.4.3 Greedy Tuning Step for Discrete Bandwidth Options

The LPs and the packing algorithm together present a method for allocating frequency intervals while avoiding fragmentation. It is designed from the outset for the case where the intervals can be arbitrarily placed. As hardware advances, eventually the hardware may achieve full flexibility in adjusting the center frequency and bandwidth. If instead only a discrete set of bandwidth options are available (as is the case in most currently available hardware), we can round the resulting assignment to comply with the available bandwidth options. In our implementation, we use an additional simple greedy tuning step in order to increase bandwidth wherever possible. The tuning step considers all the APs one by one. If for an AP there exists a wider band that is available, use it; if there is a band with lower-start position, switch to it (recall rule of thumb #2). Repeat thus iterating over all APs until no more improvement are possible.

### 4.1.5 GreedyRaising: Simple Greedy Heuristics

The LP-based approximation algorithm presented in the previous chapter provides provable performance guarantees with regard to both fairness and system throughput. In this section, we propose three simpler heuristic solutions that is both easier to deploy (it does not require solving a linear program) and, as we show in Section 4.1.8 still manages to achieve an excellent performance.

All three algorithms are based on the greedy-packing subroutine shown in Algorithm 3. This greedy packing routine takes as its input an ordering of the APs (for example, from heaviest to lightest load) and a bandwidth requirement for each AP. It then proceeds in order of the given ordering and, when considering $AP_i$, greedily attempts to pack a non-overlapping frequency interval of channel-width $B_i$ into the spectrum. As in the packing scheme of Section 4.1.4, intervals are packed at the lowest possible frequency at which the interval is non-overlapping with any previously assigned interval at a neighbor.

Depending on the given ordering and bandwidth input, the greedy-packing scheme may not succeed. If the desired channel-widths are too wide, it becomes theoretically impossible to correctly pack. However, even if it *is* theoretically possible to achieve a valid assignment of bandwidth intervals to APs, the greedy allocation may make suboptimal decisions and get stuck in the process. In this case, the subroutine returns false, thereby indicating the the caller should retry using narrower channel-widths.

The basic idea of our so-called GreedyRaising heuristics is the following. Starting from a feasible initial assignment, the heuristics "probes" APs one-by-one and checks whether greedy-packing remains successful if the AP's channel-width is raised. More specifically, GreedyRaising considers all APs in a given sequence $\mathcal{O}$. When considering an AP, its channel-width is increased to the next higher bandwidth option, and the greedy-packing subroutine is called in order

**Algorithm 3** $GreedyPack(B_1, \ldots, B_N, \mathcal{O})$ Routine

Input: Bandwidths $B_1, \ldots, B_N$ and an ordering $\mathcal{O}$ of APs
Output: If possible, a non-overlapping packing of bandwidths into the available spectrum. Return false if no packing is found.

In the order of $\mathcal{O}$: **for each** $AP_i \in V$ **do**
  pack an interval of channel-width $B_i$ in the lowest possible non-overlapping frequency.
**end for**
**if** the interval of all APs was successfully packed within the total bandwidth $[F_{bottom}, F_{top}]$ **then**
  **return** for each $AP_i \in V$ its starting frequency $S_i$ in the successful packing.
**else return** false
**end if**

**Algorithm 4** GreedyRaising Algorithm

Input: An ordering $\mathcal{O}$ of APs
Output: A non-overlapping packing of bandwidth intervals in the available spectrum.

Set parameter $\theta := 1$ and let $successful := FALSE$;
**while not** $successful$ **do**
  Let $\phi'_i := \theta \cdot D_i/(D_i + \sum_{j \in N(i)} D_j)$ for each $AP_i \in V$.
  Let $B_i$ be the largest bandwidth option s.t. $B_i \leq \phi'_i \cdot B$
  $successful := GreedyPack(B_1, \ldots, B_N, \mathcal{O})$.
  $\theta := \theta/2$;
**end for**
In the order of $\mathcal{O}$: **for each** $AP_i \in V$ **do**
  Let $\widehat{B}_i$ be the next higher bandwidth option of $B_i$.
  $successful := GreedyPack(B_1, .., \widehat{B}_i, .., B_N, \mathcal{O})$.
  **if** $successful = TRUE$ **then** $B_i := \widehat{B}_i$.
**end for**

to see whether it still succeeds. If it does, the higher bandwidth is adopted; if not, its channel-width is reset to its original value.

The only thing that remains to be defined is the ordering $\mathcal{O}$ in which the access points are considered in both the greedy packing subroutine and the main algorithm. In our studies, we distinguish three possible orderings and evaluate their relative merits. The three orderings are:

- **Most-Congested-First**: In this ordering, APs are sorted in decreasing order of their load.

- **Random**: In this ordering, APs are ordered randomly.[3]

- **Smallest-Last**: Consider an ordering $\mathcal{O}$ and let $\tau_i$ be the number of APs that are neighbors of $AP_i$ and that appear *before* $AP_i$ in $\mathcal{O}$. The smallest-last ordering is an ordering which minimizes the maximum $\tau_i$ over all APs in the network [66]. This ordering has been studied in the context of coloring problems and is based on the following observation. When considering $AP_i$ in the greedy-packing routine, $\tau_i$ reflects the number of potentially interfering intervals that have already been packed in $AP_i$'s neighborhood. Intuitively, the fewer such intervals, the easier it is to pack $AP_i$'s allocated bandwidth chunk. Considering the APs in smallest-last order minimizes the maximum obstruction that any AP faces when its bandwidth interval is packed. It has been shown in [66] that the smallest-last ordering can be computed efficiently in a single pass:

  1. $j := N$; $H := G$;
  2. Let $AP_j$ be a minimum degree AP in $H$;
  3. Remove $AP_j$ from $H$ and set $j := j - 1$;
  4. Return to step 2 until $H$ is empty;
  5. Output $\mathcal{O} = (AP_1, \ldots, AP_N)$.

As our evaluations in Section 4.1.8 will show, all three GreedyRaising heuristics have the potential of significantly outperforming the scheme based on fixed

[3]When using this ordering, we slightly adapt our heuristic in the following way. Instead of initially computing a single ordering $\mathcal{O}$ that is used throughout the procedure's execution, we generate a new random ordering $\mathcal{O}$ whenever the greedy packing subroutine is called. This reduces the risk of being stuck with a bad ordering.

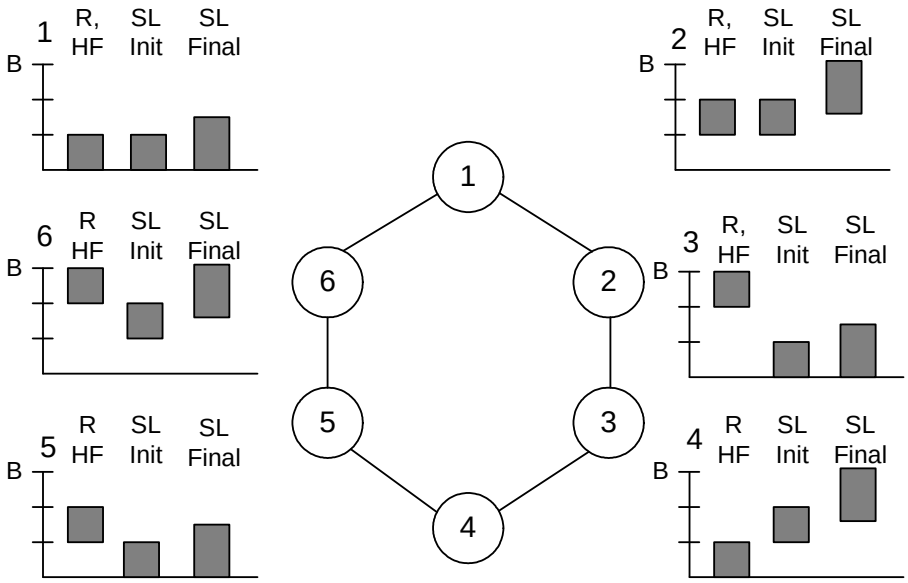


Figure 4.3: Ring network with bandwidth options $B/2$ and $B/3$ and uniform load. The smallest-last (SL) packing heuristic performs better ($L_{Sys} = 3B$) than the heavy-first (HF) and random (R) heuristics ($L_{Sys} = 2B$). In the example, the ordering of HF and R is $\mathcal{O} = (1, 4, 2, 3, 5, 6)$.

channels currently employed in IEEE 802.11. The evaluations further indicate that of the three heuristics, the one based on smallest-last orderings consistently achieves the best results.

The tendency of smallest-last to perform better than other orderings can be illustrated using simple scenarios. Consider for instance a network whose APs have (close to) uniform load and are deployed such that the resulting interference graph forms a ring (a line would yield the same results) as shown in Figure 4.3. In such a network, an optimal allocation would be to assign half of the total bandwidth to each AP, alternating between the upper and lower half. Assume that the ordering of the heavy-first and random orderings are $\mathcal{O} = (1, 4, 2, 3, 5, 6)$ (in the case of heavy-first, this can be the case if the loads are slightly different among APs, or simply by random tie-breaking). After the initial packing (Line 7 of Algorithm 4), all APs are assigned a bandwidth of $B/3$. When attempting to greedily increase some these bandwidths in the second phase, however, no fur-

ther progress is possible. In particular, regardless of which interval is increased, the packing gets stuck in the process. With the smallest-last ordering, however, the optimal allocation will be reached. Assume for instance that $AP_3$ is the first AP to be selected (possibly using a random tie-breaking rule). The next AP is one of the two having the least number of neighbors in $G \setminus \{AP_3\}$, i.e., either $AP_2$ or $AP_4$. Whichever the algorithm selects, the next AP to be selected must be one that has just one neighbor left in the graph. All possible resulting orderings therefore have the characteristics that the ring is considered "in sequence". In the initial allocation of the smallest-last ordering, every AP is allocated a bandwidth of $B/3$ as in the other heuristics (SL-Init). But, due to the efficient packing, the channel-width of all APs can be raised to the next higher bandwidth option, $B/2$.

### 4.1.6 Discussion

One of the assumptions made in our theoretical modeling is that the frequency bands assigned to neighboring APs should never overlap, which may be overly conservative in many cases [69]. However, both our model and all our algorithms can easily be adapted to incorporate co-channel interference. Particularly, if it is known how much spectrum overlap between neighboring APs is tolerable, our algorithms can be adjusted as follows. For OPT, the first two conditions of the ILP have to be adapted. In the LP-based algorithm packing algorithm it suffices to round up to a power of less than 2, and finally, the packing scheme of all our heuristic approaches will be able to pack the bandwidths more tightly. Finally, notice that both the LP-based algorithm and the GreedyRaising heuristics are computationally efficient and quickly converge to a solution even in large-scale networks.

### 4.1.7 Practical Considerations

Adaptively changing the center frequency and bandwidth allocated to an AP poses several interesting systems challenges. We need to design a new scanning mechanism for clients to discover the APs, since it might be infeasible for them to explore all possible values of center frequencies and bandwidths. Our design should be backward compatible, and the APs should also work with legacy (unmodified) Wi-Fi network cards. In this section, we present some initial thoughts on how these problems can be addressed in a real deployment.

We propose adding an extra radio to each AP, similar to a few commercially available two-radio APs [1, 67]. One radio will operate on the first channel of the band, for example channel 1 for IEEE 802.11b/g networks, or channel 36 for IEEE 802.11a networks. The other radio will adaptively adjust its center frequency and bandwidth to operate in the frequency spectrum that is not occupied by the first radio. Each AP will use the first radio to broadcast beacons and provide service to legacy clients. The beacons will also contain information about the center frequency and bandwidth of the second radio. Clients can then discover the center frequency and bandwidth of the APs by listening to beacons on the channel of the first radio. Even legacy clients will eventually go to channel 1 or 36 as part of the normal scanning process, and discover the APs.

The above architecture has multiple benefits beyond discovery and backward compatibility. For example, it enables fast handoff among clients by allowing a client to quickly discover the nearby APs, by switching to the first channel, and discovering the operating frequency and bandwidth of nearby APs (using Probe Requests and Responses).

Another practical concern is the feasibility of dynamically changing the bandwidth and central frequency of wireless cards. We are currently implementing a proof-of-concept on a wireless card based on the USB and MiniPCI Atheros

ar5523 chipset [2]. We have modified the firmware to tune the bandwidth of the wireless card to 5 MHz and 20 MHz, and change the frequency to any value in the 2.4 GHz band. To change the bandwidth, we reduced the speed of the crystal clock by tuning a register value of the Phase Locked Loop (PLL) in the firmware. Consequently, our approach requires the card to go through a firmware reset, which takes a few milliseconds. However, we strongly believe that a firmware reset is unnecessary given the evidence that the same chipset can change the bandwidth to 40 MHz using Turbo mode [2] without a firmware reset.

### 4.1.8 Performance Evaluation

In this section, we quantify the benefits of dynamic-width channels using simulations QualNet [81]. We compare our schemes, including ILP, LP, and GreedyRaising, against a recently proposed channel assignment algorithm based on fixed channels, called RaC [14]. We analyze the performance using two metrics: aggregate throughput of all clients in the WLAN and per-client fairness. The fairness metric reflects the uniformiy of throughput achieved by all clients, and we define it using Jain's fairness index: $(\sum C_i)^2/n\sum C_i^2$, where $C_i$ is the throughput obtained by client $i$, and $n$ is the total number of clients.

We first confirm the assumption that the bandwidth, and in turn throughput, achieved by an AP is proportional to the bandwidth allocated to it. We tested this assumption for two bandwidth values: 5 MHz and 20 MHz, on a Netgear AWG132 USB wireless card, which has the Atheros ar5523 chipset, with our modified firmware. In the 5 MHz case, we confirm that the data rate of the packets when the client and the AP were close to each other was 54/4 = 13.5 Mbps. The UDP throughput when using 5 MHz bandwidth was 5.9 Mbps, which is slightly less than 1/4th the throughput when using 20 MHz bandwidth (25.7 Mbps at 54 Mbps data rate). A more accurate reference clock and better frequency align-

ment mechanisms are required to further improve the effective throughput for smaller bandwidths. We believe that the advances in current radio devices, such as software defined radios [36], will greatly improve these throughput numbers.

#### 4.1.8.1 Simulation Settings

We simulate three real-world usage scenarios: a small-scale enterprise WLAN, a large enterprise/campus WLAN deployment, and a network with user mobility. For a small scale enterprise WLAN, we use the wireless usage data from [20]. This dataset contains monitoring information of 6 APs on the floor of an office building. The floorplan and location of APs is illustrated in Figure 4.4. The dataset includes the location of all the clients and their wireless usage over a 5-day work week from 8 AM to 8 PM everyday. For our simulations, we feed the coordinates of the APs and the clients in QualNet, and use our algorithms to decide the center frequency and bandwidth of each AP.

In the second set of simulations, we consider a larger enterprise network of 20 to 50 APs. We use the data from [18] that analyzed a network across three buildings comprising 177 APs to determine the number of clients associated to each AP. Since we did not have information about the clients' location, we simulate scenarios in which the associated clients are randomly placed within the transmission range of the AP.

Finally, we consider the impact of user mobility on our AP bandwidth allocation scheme. We use the model, called Model T [51], which is based on traces collected across 2 years from the large WLAN deployment in Dartmouth College, and incorporate it with the Random Waypoint Model, to model the mobility pattern of each client.

In our simulations, we study two sets of bandwidth possibilities to show the impact of bandwidth settings on our proposed approach. The first set of

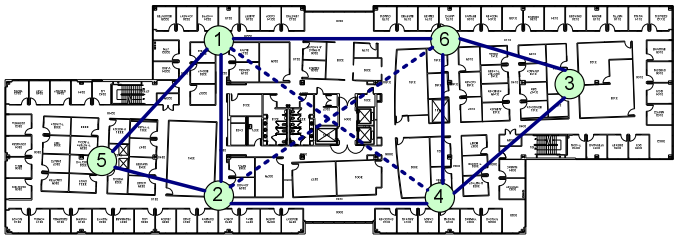


Figure 4.4: Floor plan and AP locations on the floor of an office building. The solid lines represent two interfering APs, and dashed lines indicate that the APs interfere at one of the clients.

bandwidths includes 5, 10, 20 and 40 MHz. The second set includes a wider range of bandwidths: 3, 5, 6, 7, 10, 12, 14, 20, 24, 28 and 40 MHz. We assume that each 1 MHz spectrum delivers 1.2 Mbps data rate [49]. The overall available spectrum is 86 MHz, i.e. the size of 2.4 ISM band. When using channels of 20 MHz, we have 4 non-overlapped channels. Without loss of generality, we neglect the overhead of guide band between two adjacent channels. In our proposed schemes, the clients always associate to the nearest AP and the weight of APs in our algorithms is measured by the number of clients served by the AP. In addition, to stress test the system, we set each client to have at least one backlogged CBR flow to the associated AP. The MAC layer we use is IEEE 802.11 [40]. We use the two-ray propagation model to model path loss. Furthermore, to isolate the impact of varying channel width, we assume no rate or power control.

#### 4.1.8.2 Small WLAN Deployment

We first study the effect of our scheme on a small, but real, WLAN deployment. The floorplan of the office building is illustrated in Figure 4.4. We extract the user activities from the dataset of [20]. Figure 4.5 shows the maximum number of clients that are simultaneously associated to each AP during every hour

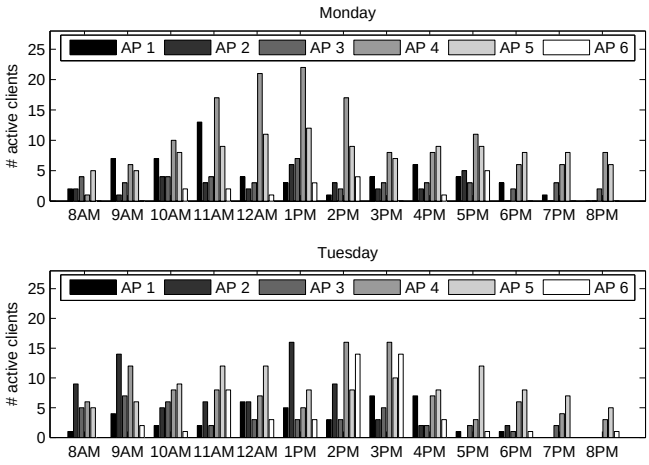


Figure 4.5: # active clients at different time of day

from 8 AM to 8 PM on Monday and Tuesday of a work week. [4] Clearly, there is a spatial and time disparity in network usage across different APs. At any given time, APs at some locations serve a significantly larger number of clients then the others. For example, from 11 AM to 2 PM on Monday, AP 4 had up to 22 clients during the peak period since it is located close to several conference rooms. Furthermore, the client populations at the APs varies significantly over time. The set of heavily-loaded APs also changes at various times of the day across different days.

Using this trace, we studied the performance of four schemes: ILP, LP, GreedyRaising (using smallest-last order), and RaC using 4 bandwidth options. Figure 4.6 depicts the throughput and the fairness index of each AP across 5 days. In all cases, ILP achieves the highest performance, up to 45% higher throughput than RaC, which is based on the fixed channels. The fairness index achieved by ILP is about 0.8, while RaC's fairness index is less than 0.5. This result shows that adaptively assigning the bandwidth to each AP not only improves the capacity of the WLAN, but also ensures more uniform service to all associated clients. On

[4]The plots for the other 3 days are omitted due to the space limitations, but they all show a similar trend.

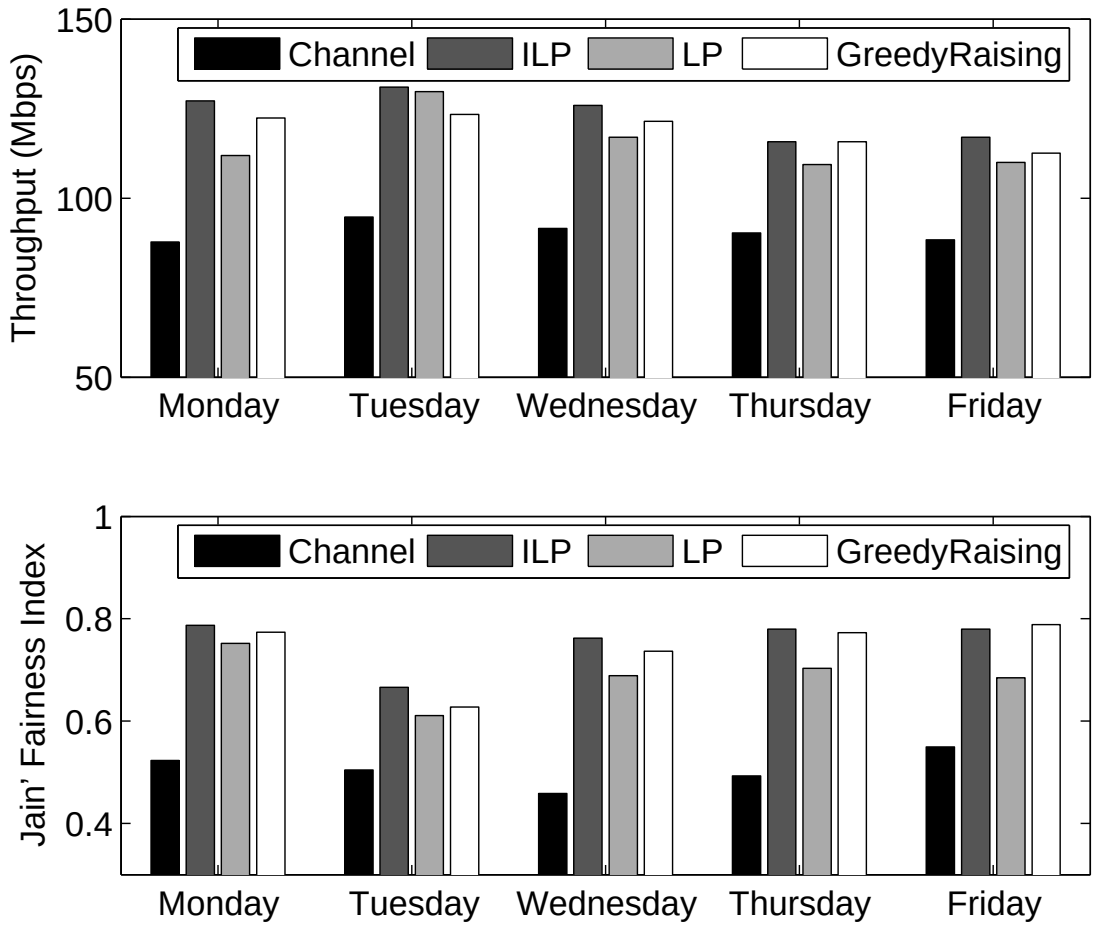


Figure 4.6: Throughput and fairness index of different allocation schemes

the other hand, when using 4 fixed channels, RaC uses coloring on the AP conflict graph, such that no two interfering APs use the same channel. However, the service received by each client is heavily biased based on their location. The clients associated to a crowded AP suffers from degraded performance, which is reflected as a suboptimal fairness index.

Compared with ILP, GreedyRaising obtains comparable performance since it emulates the operation of ILP. Based on a certain order, it attempts to raise the bandwidth for each AP starting from the initial feasible assignment. The advantage of GreedyRaising is that it is fast as it benefits from a small set of available

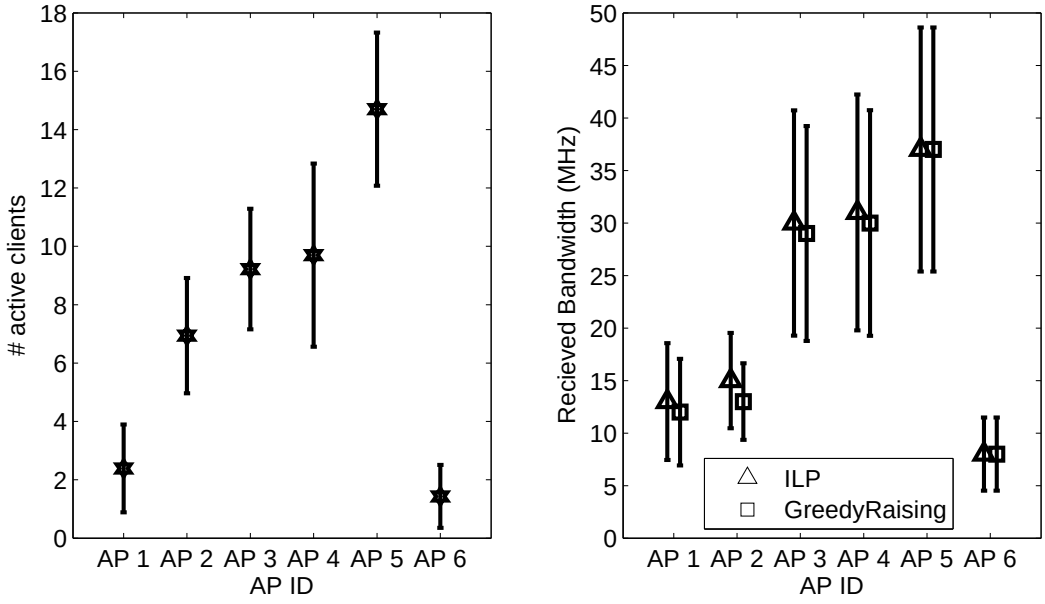


Figure 4.7: Average number of clients associated to each AP and the corresponding bandwidth allocated by our scheme.

bandwidth possibilities. The worst case complexity of the GreedyRaising algorithm is $O(n^3)$, where $n$ is the number of APs. These properties make GreedyRaising a practical solution. LP reduces the throughput by up to 14% since it evaluates all contiguous bandwidth possibilites. Consequently, it loses some throughput as it rounds the bandwidth to the nearest permissible value.

Figure 4.7 illustrates the number of clients associated to each AP and the corresponding throughput achieved by each AP. The graph shows the average and standard deviation for these values, which demonstrate that dynamic-width channels give more bandwidth to the AP that serves more clients, and the assigned bandwidth varies depending on the variance of the number of associated clients. RaC uses fixed channels, and therefore the amount of bandwidth allocated to each AP does not depend on the number of clients associated to it.

We also studied our algorithms with a larger set of bandwidth options. We observe that in this simple scenario, adding more bandwidth options does not

noticeably improve the performance. We also varied the packing schemes and compared their performance. Among them, the smallest-last scheme consistently achieve 5 –10% throughput gain compared to the other two schemes. The gain can be explained by the intuition that assigning the least congested APs last has a higher chance to fit all APs in the available spectrum.

#### 4.1.8.3 Large Wireless Networks

We now study the performance of dynamic-width channel allocation in large campus WLAN deployments. We use observation of the number of clients associated to each AP from a previous study [18]. In this trace, 50% of APs serve less than 5 users, while 10% of APs serve over 15 users. The average number of clients served by each AP is 8. Since the traces provide no information about the location of APs and clients, we randomly place the APs in a flat area of 1000 x 1000 meters. For each AP, we randomly place the client within the transmission range of the AP. The clients are assumed to be static during the experiment. We study our bandwidth allocation scheme for two different scenarios: a 20 AP WLAN and a 50 AP WLAN. For each scenario, we varied the interference among APs by changing the transmission power from -1.6 dbm to 4.2 dbm. All our results are averaged over 20 simulation runs.

Figure 4.8 illustrates the throughput and fairness index of all clients in sparse and dense deployments when using 20 APs. We emulate a sparse deployment by changing the transmit power of each AP to -1.6 dbm, such that each AP has 2 to 3 neighboring APs. In this scenario, ILP achieves 47% more throughput than RaC. This can be explained by ILP's attempt to allocate all the available bandwidth to the APs. In contrast, RaC is unable to utilize all the channels, as each AP might not have sufficient interfering neighbors. Furthermore, ILP allocates bandwidth to APs proportional to the number of clients associated to it, which further im-

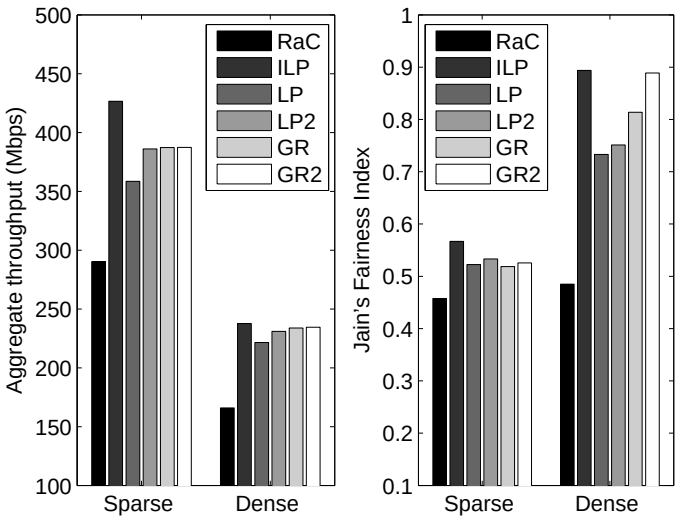


Figure 4.8: Throughput and fairness in a WLAN of 20 APs

proves system throughput. In fact, it assigns each AP with 40 MHz of bandwidth, as there is little contention among the APs. We note that the fairness index of ILP in the sparse deployment is less than 0.6 since even APs with fewer clients are allocated the maximum of 40 MHz. This appears to be the right behavior as it maximizes spectrum utilization.

We also analyzed a dense AP deployment by setting the transmission power of each AP to be 4.2 dbm (each AP has 5 to 6 interfering APs on average). In this scenario, ILP achieves 53% more throughput than RaC, and improves the fairness index to about 0.9. However, the total throughput of the system is much lesser due to increased interference. ILP allocates separate bandwidths to interfering APs, and therefore it is able to obtain better spectrum utilization. Further, since there is more contention in the system, the lightly loaded APs do not get allocated a 40 MHz bandwidth, and hence the fairness index for ILP is much higher. We note that LP and GreedyRaising obtain near optimal throughput.

We now compare the GreedyRaising algorithm with RaC in a larger WLAN of 50 APs. As we see in Figure 4.9, the system throughput achieved by GreedyRais-

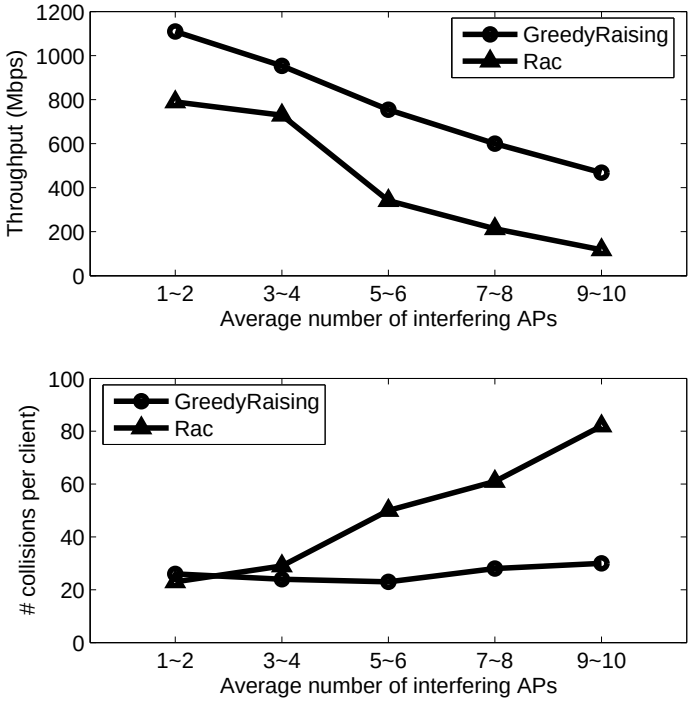


Figure 4.9: System throughput and per-client collisions in a WLAN of 50 APs

ing and RaC decreases with increased interference among APs. However, GreedyRaising gets much higher throughput. This can be explained by the second graph, which plots the number of collisions per client with an increase in the number of interfering APs. GreedyRaising allocates separate chunks of the spectrum to interfering APs, and hence the number of per-client collisions stays the same. However, there are not enough non-overlapping channels available to RaC, and hence increased interference among APs increases the number of collisions at each client.

#### 4.1.8.4 Handling User Mobility

Given the recent growth of mobile applications, such as VoIP, over WLANs, we study the effectiveness of our approach in handling user mobility. We stress test our system by having 40% of wireless clients mobile. We combine the regis-

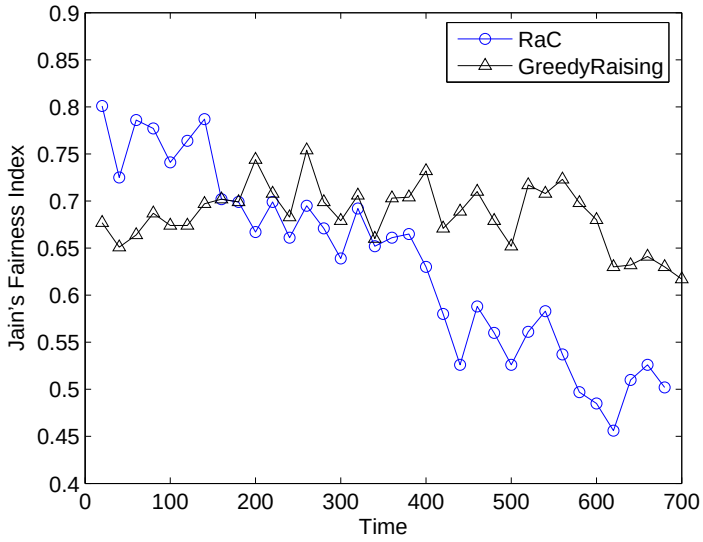


Figure 4.10: Fairness Index of 160 clients in a 25 AP WLAN when aggregated over 20 second intervals over 700 seconds.

tration and mobility pattern defined in Model T [51] with the Random Waypoint Model. Model T captures the popular APs towards which most of the client movements are directed. Each node selects an AP using Model T, and moves towards it with a speed chosen randomly from an interval, ($V_{min}$, $V_{max}$]. Upon reaching its destination, the node moves to a new destination after it pauses for a random period between 0 and 10 seconds. We set $V_{min}$ at 0.01 m/s and vary $V_{max}$ from 0.2 to 1.2 m/s.

We consider a WLAN with 25 APs deployed uniformly in a 500 m x 500 m area. The transmission power of each AP set to -1.6 dbm, which gives an approximation transmission range of 100 meters. The number of interfering APs varies from 3 to 8. Initially, clients are uniformally distributed across each AP. At the start of the simulation, clients begin to move towards the APs defined by model T. RaC reassigns the channel every 50 seconds. GreedyRaising adjusts the bandwidth allocation if a new assignment improves the fairness index or the system throughput by more than 10%.

Figure 4.10 shows the fairness index for the WLAN over time. Each point

in the graph is an aggregate fairness value computed over a 20 second interval. Initially, GreedyRaising has a worse fairness index than RaC. This is because GreedyRaising assigns more bandwidth to APs on the edge, and lesser bandwidth to APs in the middle to create enough channels. As clients begin to move, their distribution across all the APs gets skewed as the popular APs serve a much larger number of clients. GreedyRaising captures this change and dynamically adjusts the channel widths, therefore, achieving consistent fairness over time. On the other hand, RaC is based on the fixed channels, and consequently it is unable to handle skewed client AP distributions.

We also measured the overall system throughput aggregated over the entire 700 second interval. Our approach delivers a total throughput of 273 Mbps while RaC delivers 195 Mbps throughput. The reason for the difference is similar to observations in the previous subsections.

### 4.1.9 Related Work

AP load balancing in WLANs attempts to evenly distribute the number of clients across all APs in a region. One way to solve this problem is to use Cell Breathing [17]. In this approach the APs in a region adjust their transmission power to force some of its' associated clients to handoff to neighboring APs. Similarly, APs might also increase their transmission power to induce clients to associate to them. This technique is very useful in hotspot and flash crowd scenarios, where many users associate to the same AP, even when the neighboring APs are lightly loaded. Although this scheme is useful is balancing the load across APs, it can potentially worsen the performance if clients associate to far away APs and send the packets at a lower data rate. An alternate approach is client-based, where Wi-Fi devices take smart decisions and associate to the more lightly loaded AP [87]. However, this scheme does not completely solve the user unfair-

ness problem. For example, many clients close to an AP might be contending for resources on a channel of fixed bandwidth, while fewer clients on a neighboring AP might be contending on the same bandwidth.

Another approach to solving the user unfairness problem is to assign more APs to the WLAN [13, 68]. Each user will have a dedicated AP in most scenarios, and so every user gets around the same throughput. However, the benefits of these schemes are limited because of fixed bandwidth channels. First, in extremely dense hotspots, the number of clients might outnumber the number of APs and user unfairness might be unavoidable. Second, when there are very few clients in the network, this technique will waste a large amount of bandwidth.

We overcome the shortcomings of the above schemes by attacking the fundamental problem of fixed-width channels. We allocate variable size contiguous spectrum to the APs as a function of its load. There are several schemes that are complementary to ours and can be integrated with our proposed approach to further enhance the performance of the WLAN. For example, each AP may allocate a different spectrum slice to every client that is associated to it. This will minimize interference from nearby transmitters and give better throughput.

## 4.2 Scalable Medium Access Control

In this section, we present the two-tier design of TMAC framework, which incorporates centralized, coarse-grained regulation at the higher tier and distributed, fine-grained channel access at the lower tier. Token-coordinated channel regulation provides coarse-grained coordination for bounding the number of contending stations at any time. It effectively controls the contention intensity and scales to various population sizes. Adaptive distributed channel access at the lower tier exploits the wide range of high data rates via the adaptive ser-

vice model. It opportunistically favors stations under better channel conditions, while ensuring each station a adjustable fraction of the channel time based upon the perceived channel quality. These two components work together to address the scalability issues. We present these two components in Section 4.2.1 and Section 4.2.2, respectively.

### 4.2.1 Token-coordinated Channel Regulation

#### 4.2.1.1 Higher-Tier Design Overview

TMAC employs a simple token mechanism in regulating channel access at the coarse-time scale (e.g., in the order of 30 $\sim$ 100 $ms$). The goal is to significantly reduce the intensity of channel contention incurred by a large population of active stations. The base design of the token mechanism is motivated by the observation that polling-based MAC works more efficiently under heavy network load [88, 96], while random contention algorithms better serve bursty data traffic under low load conditions [9, 34]. The higher-tier design, therefore, applies a polling model to multiplex traffic loads of stations within the token group.

Figure 4.11 schematically illustrates the token mechanism in TMAC. An AP maintains a set of associated stations, $S = \{s_1, s_2, ..., s_n\}$, and organizes them into $g$ number of disjoint token groups, denoted as $V_1, V_2, ..., V_g$. Apparently, $\bigcup_{i=1}^{g} V_i = S$, and $V_j \cap V_j = \emptyset$ ( $1 \leq i, j \leq g$ and $i \neq j$). Each token group, assigned a unique Token Group ID (TGID), accommodates a small number of stations, $N_{V_i}$, and $N_{V_i} \leq \bar{N}_V$, where $\bar{N}_V$ is a pre-defined upper bound. The AP regularly distributes a token to an eligible group, within which the stations contend for channel access via the enhanced random channel procedure in the lower tier. The period during which a given token group $V_k$ obtains service is called *token service period*, denoted by $TSP_k$, and the transition period between two con-

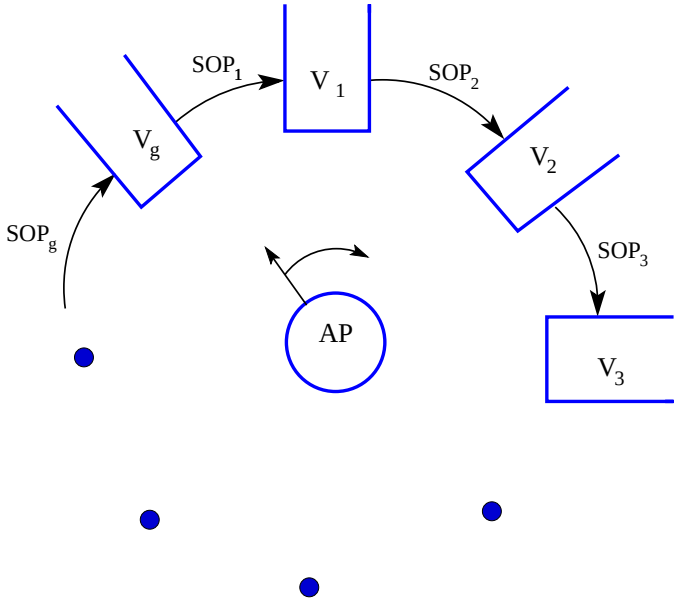


Figure 4.11: Token distribution model in TMAC

secutive token groups is the *switch-over period*. The token service time for a token group $V_k$ is derived using: $TSP_k = (N_{V_k}/\bar{N}_V)\overline{TSP}, (1 \leq k \leq g)$, where $\overline{TSP}$ represents the maximum token service time. Upon the timeouts of $TSP_k$, the AP grants channel access to the next token group $V_{k+1}$.

To switch between token groups, the higher-tier design constructs a token distribution packet (TDP), and broadcasts it to all stations. The format of TDP, shown in Figure 4.12, is compliant with the management frame defined in 802.11b. In each TDP, a timestamp is incorporated for time synchronization, $g$ denotes the total number of token groups, and the token is allocated to the token group specified by the TGID field. Within the token group, contending stations use $CW_t$ in random backoff. The $R_f$ and $T_f$ fields provide two design parameters employed by the lower tier. The optional field of group member IDs is used to perform membership management of token groups, which can be MAC addresses, or dynamic addresses [90] in order to reduce the addressing overhead. The length of TDP ranges from 40 to 60 bytes ($\overline{N}_V = 20$, each ID uses 1 byte), taking less than $100us$ at 6Mbps rate. To reduce the token loss, TDP is typically

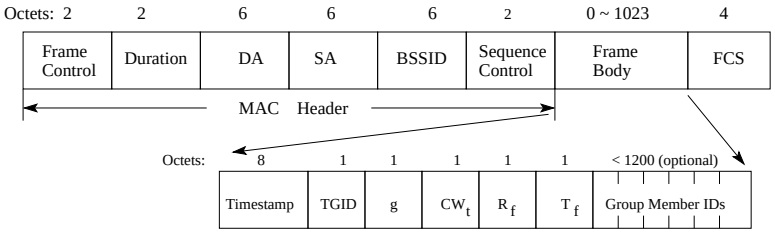


Figure 4.12: Frame format of Token Distribution Packet

transmitted at the lowest rate.

We need to address three concrete issues to make the above token operations work in practice, including membership management of token groups, policy of scheduling the access group, and handling transient conditions (e.g., when TDP is lost).

#### 4.2.1.2 Membership Management of Token Groups

When a station joins the network, TMAC assigns it to an eligible group, then piggybacks $TGID$ of the token group in the association response packet [75], along with a local ID [90] generated for the station. The station records the $TGID$ and the local ID received from the AP. Once a station sends a de-association message, the AP simply deletes the station from its token group. The groups are reorganized if necessary. For performing membership management, the AP generates a TDP carrying the optional field that lists IDs of current members in the token group. Upon receiving the TDP with the ID field, each station with a matched $TGID$ purges its local $TGID$. The station, whose ID appears in the ID field, extracts the $TGID$ value from the TDP and updates its local $TGID$.

The specific management functions are described in the pseudo code listed in Algorithm 5. Note that we evenly split a randomly chosen token group if all the groups contain $\bar{N}_V$ stations, and merge two token groups if necessary. In this way, we keep the size of token group above $\bar{N}_V/4$ to maximize the benefits from traffic load multiplexing. Other optimizations can be further incorporated

into the management functions. At present, we keep the current algorithm for simplicity.

**Algorithm 5** Group Membership Management Functions

*Function 1: On station $s$ joining the network*
**if** $g$ == 0 **then**
  create the token group $V_1$ with $TGID_1$
  $V_1 = s$, set the update bit of $V_1$
**else**
  search for $V_i, s.t., N_{V_i} < \bar{N}_v$,
  **if** $V_i$ exists **then**
    $V_i = V_i \cup s$,set the update bit of $V_i$
  **else**
    randomly select a token group $V_i$
    Split $V_i$ evenly into two token groups, $V_i, V_{g+1}$
    $V_i = V_i \cup s$
    set the update bit of $V_i$ and $V_{g+1}$, g = g + 1
  **end if**
**end if**
*Function 2: On station $s$, $s \in V_i$, leaving the network*
$V_i = V_i - s$
**if** $N_{V_i} == 0$ **then**
  delete $V_i$, reclaim $TGID_i$, g = g - 1
**end if**
**if** $N_{V_i} < \bar{N}_v/4$ **then**
  search for $V_j, s.t., N_{V_j} < \bar{N}_v/2$,
  **if** $V_j$ exists **then**
    $V_j = V_j \cup V_i$
    delete $V_i$, reclaim $TGID_i$
    set the update bit of $V_j$, g = g - 1
  **end if**
**end if**

#### 4.2.1.3 Scheduling Token Groups

Scheduling token groups deals with the issues of setting the duration of $\overline{TSP}$ and the sequence of the token distribution.

The $\overline{TSP}$ is chosen to strike a balance between the system throughput and the delay. In principle, the size of the $\overline{TSP}$ should allow for every station in a

token group to transmit once for a period of its temporal share $T_i$. $T_i$ is defined in the lower-tier design and typically in the order of several milliseconds. The network throughput performance improves when $T_i$ increases [97]. However, increasing $T_i$ enlarges the token circulation period, $g * \overline{TSP}$, thus affecting the delay performance. Consequently, $\overline{TSP}$ is a tunable parameter in practice, depending on the actual requirements of throughput/delay. The simulation results of Section VI provide more insights of selecting a proper $\overline{TSP}$.

To determine the scheduling sequence of token groups, TMAC uses a simple round-robin scheduler to cyclicly distribute the token among groups. It treats all the token groups with identical priority.

#### 4.2.1.4 Handling Transient Conditions

Transient conditions include the variation in the number of active stations, loss of token messages, and stations with abnormal behaviors.

The number of active stations at an AP may fluctuate significantly due to bursty traffic load, roaming, and power-saving schemes [96] [29]. TMAC exploits a token-based scheme to limit the intensity of spatial contention and collisions. However, potential channel wastage may be incurred due to under-utilization of the allocated $TSP$ when the number of active stations sharply changes. TMAC takes a simple approach to adjust the TSP boundary. The AP announces the new TGID for the next group after deferring for a time period $TIFS = (DIFS + m * \overline{CW}_t * \sigma)$, where $\overline{CW}_t$ is the largest CW in the current token group, $m$ is the maximum backoff stage, and $\sigma$ is the mini-slot time unit (i.e., $9\mu s$ in 802.11a). The lower-tier operation in TMAC ensures that TIFS is the maximum possible backoff time. In addition, if a station stays in the idle status longer than the defined idle threshold, the AP assumes that it enters the power-saving mode, records it in the idle station list, and performs the corresponding management function for a

leaving station. When new traffic arrives, the idle station executes the routine defined in the second transient condition to acquire a valid TGID, and then returns to the network.

Under the second transient condition, a station may lose its transmission opportunity in a recent token service period or fail to update its membership due to TDP loss. In this scenario, there are two cases. First, if the lost TDP message informs group splitting, the station belonging to the newly generated group, continues to join TSP matches its original TGID. The AP, upon detecting this behavior, unicasts the station with the valid TGID to notify its new membership. Second, if the lost TDP message announces group merging, the merged stations may not be able to contend for the channel without the recently assigned TGID. To retrieve the valid TGID, each merged station sends out reassociation/reauthentication messages after timeouts of $g * \overline{TSP}$.

We next consider the station with abnormal behaviors, i.e., the station transmits during the TSP that it does not belong to. Upon detecting the abnormal activities, the AP first re-assigns it to a token group if the station is in the idle station list. Next, a valid TGID is sent to the station to compensate the potentially missed TDP. If the station continues the behavior, the AP can exclude the station by transmitting it a de-association message.

### 4.2.2 Adaptive Distributed Channel Access

The lower-tier design addresses the issues of capacity scalability and protocol overhead scalability in high-speed wireless LANs with an Adaptive Service Model (ASM). The proposed ASM largely reduces channel access overhead and offers differentiated services that can be *adaptively* tuned to leverage high rates of stations. The following three subsections describe the contention mechanism, the adaptive channel sharing model, and the implementation of the model.

#### 4.2.2.1 Channel Contention Mechanism

Channel contention among stations within an eligible token group follows the carrier sensing and random backoff routines defined in DCF [75, 77] mechanism. Specifically, a station with pending packets defers for a DIFS interval upon sensing an idle channel. A random backoff value is then chosen from $(0, \overline{CW}_t)$. Once the associated backoff timer expires, RTS/CTS handshake takes place, followed by DATA transmissions for a time duration specified by ASM. Each station is allowed to transmit *once* within a given token service period to ensure the validity of ASM among stations across token groups. Furthermore, assuming most of stations within the group are active, AP can estimate the optimal value of $\overline{CW}_t$ based on the size of the token group, which will be carried in the $CW_t$ field of TDP messages. $\overline{CW}_t$ is derived based on the results of [34]:

$$\overline{CW}_t = \frac{2}{\zeta(1 + p\Sigma_{i=0}^{m-1}(2p)^i)} \tag{4.8}$$

where $p = 1-(1-\zeta)^{n-1}$ and the optimal transmission probability $\zeta$ can be explicitly computed using $\zeta = 1/(\bar{N}_V \cdot \sqrt{T_c^*/2})$, and $T_c^* = (RTS + DIFS + \delta)/\sigma$. $m$ denotes the maximum backoff stage, which has marginal effect on system throughput with RTS/CTS turned on [34], and $m$ is set to 2 in TMAC.

#### 4.2.2.2 Adaptive Service Model

The adaptive sharing model adopted by TMAC extracts the multiuser diversity by granting the users under good channel condition proportionally longer transmission durations. In contrast, the state-of-the-art wireless MACs do not adjust the time share to the perceived channel quality, granting stations with either identical throughput share [75] or equal temporal share [16, 26, 43], under ideal-

ized conditions. Consequently, the overall network throughput is significantly reduced since these MAC schemes ignore the channel conditions when specifying the channel sharing model. ASM works as follows. The truncated function (2) is exploited to define the service time $T_{ASM}$ for station $i$, which transmits at the rate of $r_i$ upon winning the channel contention:

$$T_{ASM}(r_i) = \begin{cases} (r_i/R_f)T_f & r_i \geq R_f \\ T_f & r_i < R_f \end{cases} \tag{4.9}$$

The model differentiates these two classes of stations, high-rate and low-rate stations, by defining the reference parameters, namely, the reference transmission rate $R_f$ and the reference time duration $T_f$. Stations with transmission rates higher than or equal to $R_f$ are categorized as high-rate stations, thus granted *proportional temporal share* in that the access time is roughly proportional to the current data rate. For low-rate stations, each of them is provided *equal temporal share* in terms of identical channel access time $T_f$. Thus, ASM awards high-rate stations with a proportional longer time share and provides low-rate stations equal channel shares. In addition, the current DCF and OAR MAC become the specific instantiations of ASM by tuning the reference parameters.

#### 4.2.2.3 Implementation Using Adaptive Batch Transmission and Block ACK

To realize ASM, AP regularly advertises the two reference parameters $R_f$ and $T_f$ within a TDP. Upon receiving TDP, stations in the matched token group extract the $R_f$ and $T_f$ parameters, and contend for the channel access. Once a station succeeds in contention, adaptive batch transmission allows for the station to transmit multiple concatenated packets for a period equal to the time share com-

puted by ASM. The adaptive batch transmission can be implemented at either the MAC layer as proposed in OAR [16] or the physical layer as in MAD [102]. To further reduce protocol overhead at the MAC layer, we exploit the block ACK technique to acknowledge $A_f$ number of back-to-back transmitted packets in a single Block-ACK message, instead of per-packet ACK in the 802.11 MAC. The reference parameter $A_f$ is negotiated between two communicating stations within the received-based rate adaptation mechanism [8] by utilizing RTS/CTS handshake.

### 4.2.3 Performance Analysis

In this section, we analyze the scalable performance obtained by TMAC in high-speed wireless LANs, under various user populations. We first characterize the overall network throughput performance in TMAC, then analytically compare the gain achieved by ASM with existing schemes. Also, we provide analysis on the three key aspects of scalability in TMAC.

#### 4.2.3.1 Network Throughput

To derive the network throughput in TMAC, let us consider a generic network model where all $n$ stations are randomly located in a service area $\Omega$ centered around AP, and stations in the token groups always have backlogged queues of packets at length $L$. Without loss of generality, we assume each token group accommodates $N_V$ number of active stations, and there are total $g$ groups. We ignore the token distribution overhead, which is negligible compared to the TSP duration. Thus, the expected throughput $S_{TMAC}$ can be derived based on the results from [61].

$$S_{TMAC} = \frac{P_t \cdot E[P]}{P_t \cdot T_t + P_c \cdot T_c + P_i \cdot T_{SLOT}} \tag{4.10}$$

$$P_t = N_V \cdot \zeta \cdot (1-\zeta)^{N_V - 1} \tag{4.11}$$

$$P_c = 1 - P_t - P_i \tag{4.12}$$

$$P_i = (1-\zeta)^{N_V} \tag{4.13}$$

$$\zeta = \frac{2}{1 + CW_{min} + p \cdot CW_{min} \sum_{i=0}^{m-1} (2p)^i} \tag{4.14}$$

$$p = 1 - (1-\zeta)^{N_V} \tag{4.15}$$

$E[P]$ is the expected payload size; $T_c$ is the average time the channel is sensed busy by stations due to collisions; $T_s$ denotes the duration of busy channel in successful transmissions. Suppose that the physical layer offers $M$ options of the data rates as $r_1, r_2, \cdots, r_M$, and $P(r_i)$ is the probability that a node transmits at rate $r_i$. When TMAC adopts the adaptive batch transmission at the MAC layer, the values of $E[P]$, $T_c$, and $T_s$ are expressed as follows.

$$E[P] = \sum_{i=1}^{M} P(r_m) \cdot L \cdot \frac{T_{ASM}(r_i)}{T^{EX}(r_i)}$$

$$T_c = T_{DIFS} + T_{RTS} + \delta$$

$$T_s = T_c + T_{CTS} + \sum_{i=1}^{M} P(r_i) T_{ASM}(r_i) + T_{SIFS} + 2\delta$$

$T^{EX}(r_i)$ is the time duration of the data packet exchange at rate $r_i$, specified by $T^{EX}(r_i) = T_{PH} + T_{MH} + L/r_i + 2 \cdot T_{SIFS} + T_{ACK}$, with $T_{PH}$, $T_{MH}$ being the overhead of physical-layer header and MAC-layer header, respectively. $\delta$ is the propagation delay.

Next, based on the above derivations and results in [16, 34], we compare the network throughput obtained with TMAC, DCF, OAR. The parameters used

Table 4.1: Comparison of TMAC, DCF, and OAR

| | Analysis | | | Simulation | |
|---|---|---|---|---|---|
| | S(Mbps) | $T_s(\mu s)$ | $E[P]$(bits) | S(Mbps) | $S_f$(Mbps) |
| DCF MAC | 18.41 | 404.90 | 8192 | 18.79 | 20.24 |
| OAR MAC | 31.50 | 781.24 | 20760 | 32.11 | 26.52 |
| $TMAC_{R_f=108}$ | 38.46 | 2119.42 | 83039 | 38.92 | 39.31 |
| $TMAC_{R_f=54}$ | 41.64 | 1763.27 | 75093 | 42.13 | 42.59 |
| $\text{TMAC}_{R_f=24}$ | 46.31 | 1341.61 | 62587 | 46.85 | 47.37 |

to generate the numerical results are chosen as follows: n is 15; $g$ is 1 and $L$ is $1K$; $T_f$ is set to 2ms; the series of possible rates are 24, 36, 54, 108, and 216 in Mbps, among which a station uses each rate with equal probability; other parameters are listed in Table 4.3. The results from numerical analysis and simulation experiments are shown in Table 4.1 as the $R_f$ parameter in ASM of TMAC varies. Note that TMAC, with $R_f$ set to $108Mbps$, improves the transmission efficiency, measured with $S_f = E[P]/T_s$, by 22% over OAR. On further reducing $R_f$, the high-rate stations are granted with the proportional higher temporal share. Therefore, TMAC with $R_f = 24Mbps$ achieves 48% improvement in network throughput over OAR, and 84% over DCF. Such throughput improvements demonstrate the effectiveness of ASM by leveraging high data rates perceived by multiple stations.

#### 4.2.3.2 Adaptive Channel Sharing

Here, we analyze the expected throughput of ASM, exploited in the lower tier of TMAC, as compared with those of the equal temporal share model proposed in OAR [16] and of the equal throughput model adopted in DCF [75].

Let $\phi_i^{ASM}$, $\phi_i^{OAR}$ be the fraction of time that station $i$ transmits at rate $r_i$ during a time duration $T$, where $0 \leq \phi_i \leq 1$. During the interval $T$, $n$ denotes the

number of stations in the equal temporal sharing policy, and $n'$ is the number of stations transmitting within the adaptive service model, clearly $n' \geq n$ Then, we have the following equality:

$$\sum_{i=1}^{n} \phi_i^{OAR} = \sum_{i=1}^{n'} \phi_i^{ASM} = 1 \tag{4.16}$$

Therefore, the expected throughput achieved in ASM is given by:$S_{ASM} = \sum_{i=1}^{n'} r_i \phi_i^{ASM}$. We obtain the following result, using the above notations:

**Proposition 4.2.1.** *$S_{ASM}$, $S_{OAR}$ and $S_{DCF}$ are the total expected throughput attained by ASM, OAR and DCF, respectively. We have:*

$$S_{ASM} \geq S_{OAR} \geq S_{OAR}$$

*Proof* From the concept of equal temporal share, we have $\phi_i^{OAR} = \phi_j^{OAR}$, ($1 \leq i, j \leq n$). The expected throughput in equal temporal share is derived as:

$$S_{OAR} = \sum_{i=1}^{n} r_i \phi_i^{OAR} = \frac{1}{n} * \sum_{i=1}^{n} r_i.$$

Thus, by relations (4.16) and Chebyshev's sum inequality, we can have the following result.

$$S_{OAR} \leq \frac{1}{n} \sum_{i=1}^{n} \phi_i^{ASM} \sum_{i=1}^{n} r_i \leq \sum_{i=1}^{n} \phi_i^{ASM} r_i \leq S_{ASM}$$

Similarly, we can show that $S_{DCF} \leq S_{OAR}$. □

#### 4.2.3.3 Performance Scalability

We analytically study the scalability properties achieved by TMAC, while we show that the legacy solutions do not possess such appealing features.

**Scaling to user population:** It is easy to show that TMAC scales to the user populations. From the throughput characterization of (4.10)∼(4.15), we observe that the throughput of TMAC is only dependent on the token group size $N_V$, instead of the total number of users $n$. Therefore, the network throughput in TMAC scales with respect to the total number of stations $n$.

To demonstrate the scalability constraints of the legacy MAC, we examine the DCF with RTS/CTS handshakes. Note that DCF can be viewed as a special case of TMAC, in which all $n$ stations stay in the same group, thus $N_V = n$. We measure two variables of $\zeta$ and $T_w$. $\zeta$ is the transmission probability of a station at a randomly chosen time slot and can be solved using the equations of (4.14) and (4.15) with $N_V = n$. $T_W$ denotes the time wasted on the channel due to collisions per successful packet transmission, and can be computed by:

$$T_W = (T_{DIFS} + T_{RTS} + \delta)(\frac{1-(1-\zeta)^n}{n\zeta(1-\zeta)^{n-1}} - 1),$$

where $\delta$ denotes the propagation delay.

As the number of stations increases, the values of $\zeta$ and $T_W$ in the DCF are listed in Table 4.2 and the network throughput is shown in Figure 2.5.b. Although $\zeta$ decreases as the user size expands because of the enlarged CW in exponential backoff, the channel time wasted in collisions, measured by $T_W$, increases almost linearly with $n$. The considerable wastage of channel time on collisions leads to approximately 50% network throughput degradation as the user size reaches 300, as shown by simulations.

**Scaling of protocol overhead and physical-layer capacity:** Within a token group, we examine the protocol overhead at the lower tier as compared to DCF. At a given data rate $r$, the protocol overhead $T_o$ denotes the time duration of executing the protocol procedures in successfully transmitting a $E[P]$-bytes packet,

Table 4.2: Analysis results for $\zeta$ and $T_W$ in DCF

| $n$ | 15 | 45 | 105 | 150 | 210 | 300 |
|---|---|---|---|---|---|---|
| $\zeta$ | 0.0316 | 0.0177 | 0.0110 | 0.0090 | 0.0075 | 0.0063 |
| $T_W(\mu s)$ | 21.80 | 43.24 | 72.78 | 92.75 | 119.61 | 163.34 |

which is given by:

$$T_o^{DCF} = T_o^p + T_{idle} + T_{col}, \tag{4.17}$$

$$T_o^{ASM} = \frac{T_o^{DCF}}{B_f} + T_o^{EX}. \tag{4.18}$$

$T_{idle}$ and $T_{col}$ represent the amount of idle time and the time wasted on collisions for each successful packet transmission, respectively. $T_o^p$ specifies in DCF the protocol overhead spent on every packet, which is equal to $(T_{RTS} + T_{CTS} + T_{DIFS} + 3T_{SIFS} + T_{ACK} + T_{PH} + T_{MH})$. $T_o^{EX}$ denotes the per-packet overhead of the adaptive batch transmission in ASM, which is calculated by $(2T_{SIFS} + T_{ACK} + T_{PH} + T_{MH})$. $B_f$ is the number of packets transmitted in $T_{ASM}$ interval and $B_f = T_{ASM}/T^{EX}$. From (4.17) and (4.18), we note that the protocol overhead in ASM is reduced by the factor of $B_f$ as compared with DCF, and $B_f$ is a monotonically increasing function of data rate $r$. Therefore, TMAC effectively controls its protocol overhead and scales to the channel capacity increase, while DCF suffers from fixed per-packet overhead, throttling the scalability of its network throughput. Moreover, $T_o^{EX}$ is the fixed overhead in TMAC, incurred by physical-layer preambles, inter-frame spacings, and protocol headers. It is the major constraint to further improve the throughput in the MAC layer.

**Scaling to physical-layer capacity:** To demonstrate the scalability achieved by TMAC with respect to the channel capacity $R$, we rewrite the network through-

put as the function of $R$, and obtain

$$S_{DCF} = \frac{L}{R \cdot T_o^{DCF} + L} \cdot R \tag{4.19}$$

$$S_{TMAC} = \frac{L}{(\frac{T_o^{DCF}}{T_{ASM}} + 1)(R \cdot T_o^{EX} + L)} \cdot R \tag{4.20}$$

Note that $T_{ASM}$ is typically chosen in the order of several milliseconds, thus having $T_{ASM} \gg T_o^{DCF}$. Now, the limiting factor of network throughput is $L/(R \cdot T_o^{DCF})$ in DCF, and $L/(R \cdot T_o^{EX})$ in ASM. Since $T_o^{EX} \ll T_o^{DCF}$ and $T_o^{EX}$ is in the order of hundreds of microseconds (for example, $T_o^{EX} = 136\mu s$ in 802.11a/n), ASM achieves much better scalability as $R$ increases, while the throughput obtained in DCF is restrained by the increasingly enlarged overhead ratio. In addition, the study shows transmitting packets at larger size $L$ can greatly improve network throughput. Therefore, the technique of packet aggregation at the MAC layer and payload concatenation at the physical layer is promising in next-generation high-speed wireless LANs.

### 4.2.4 Simulation Study

We conduct extensive simulation experiments to evaluate scalability performance, channel efficiency and sharing features achieved by TMAC in wireless LANs. Five environment parameters are varied in the simulations to study TMAC's performance, including user population, physical-layer rate, traffic type, channel fading model, and fluctuations in the number of action stations. Two design parameters, $T_f$ and $A_f$, are investigated to quantify their effects ($R_f$ has been examined in the previous section). We also plot the performance of the legacy MACs, 802.11 DCF and OAR, in demonstrating their scaling constraints. We use TMAC$_{DCF}$ and TMAC$_{OAR}$ to denote TMAC employing DCF or OAR in the lower tier, which are both specific cases of TMAC,

Table 4.3: PHY/MAC parameters used in the simulations

| SIFS | 16$\mu$s | DIFS | 34$\mu$s |
|---|---|---|---|
| Slot Time | 9$\mu$s | PIFS | 25 $\mu$s |
| ACK size | 14 bytes | MAC Header | 34 bytes |
| Peak DataRate (11a) | 54Mbps | Basic DataRate (11a) | 6Mbps |
| Peak DataRate (11n) | 216Mbps | Basic DataRate (11n) | 24Mbps |
| PLCP Preamble | 16$\mu$s | PLCP Header Length | 24bytes |

The simulation experiments are conducted in *ns-2* with the extensions of Ricean channel fading model [84] and the receive-based rate adaptation mechanism [8]. Table 4.3 lists the parameters used in the simulations based on IEEE 802.11b/a [42, 75] and the leading proposal for 802.11n [7]. The transmission power and radio sensitivities of various data rates are configured according to the manufacturer specifications [3] and 802.11n proposal [7]. The following parameters are used, unless explicitly specified. Each token group has 15 stations. $T_f$ allows $2ms$ batch transmissions at MAC layer. Each block ACK is sent for every two packets (i.e. $A_f = 2$). Any packet loss triggers retransmission of two packets. Token is announced approximately every $35ms$ to regulate channel access. Each station generates constant-bit-rate traffic, with the packet size set to $1K$ bytes.

#### 4.2.4.1 Scaling to User Population

We first examine the scalability of TMAC in aspects of network throughput and average delay as population size varies. **Network throughput:** Figure 4.13 shows that both TMAC$_{ASM}$ and TMAC$_{OAR}$ achieves scalable throughput, experiencing less than 6% throughput degradation, as the population size varies from 15 to 315. In contrast, the network throughput obtained with DCF and OAR does not scale: the throughput of DCF decreases by 45.9% and 56.7% at the rates of

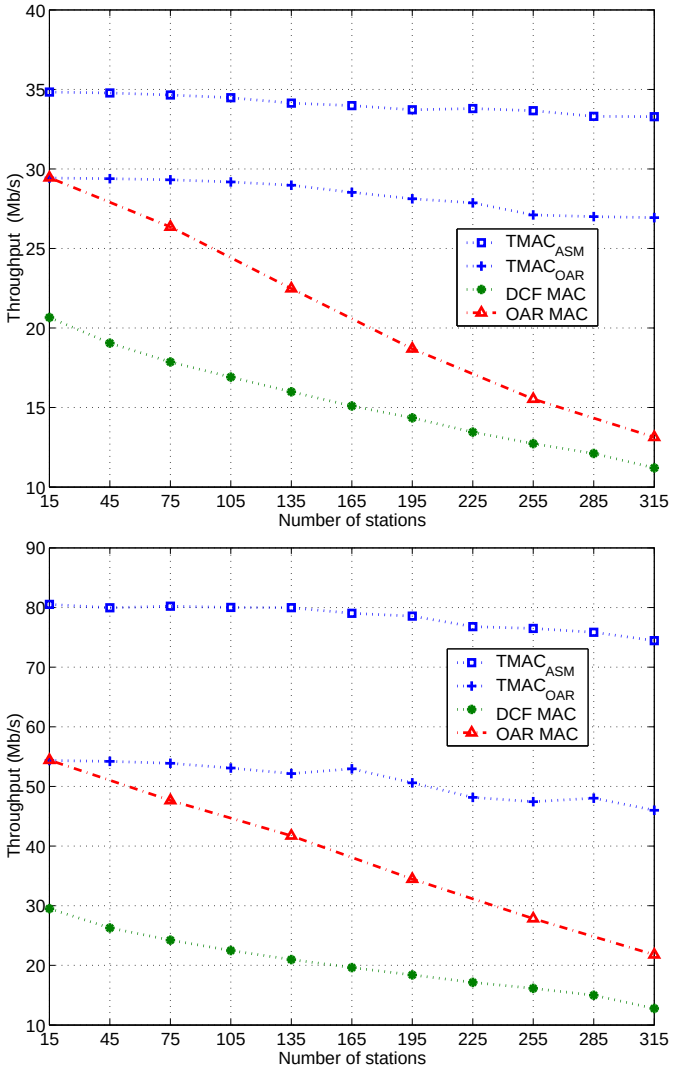


Figure 4.13: Network throughput vs. the number of stations at at 54Mbps and 216Mbps link capacity

Table 4.4: Average delay (in second) at 216Mbps

| Num. | 15 | 45 | 75 | 135 | 165 | 225 | 285 |
|---|---|---|---|---|---|---|---|
| DCF MAC | 0.165 | 0.570 | 0.927 | 1.961 | 3.435 | 4.539 | 5.710 |
| TMAC$_{DCF}$ | 0.163 | 0.822 | 1.039 | 1.654 | 2.400 | 2.590 | 2.870 |
| TMAC$_{ASM}$ | 0.053 | 0.169 | 0.359 | 0.620 | 0.760 | 0.829 | 1.037 |

54Mbps and 216Mbps, respectively, and the throughput in OAR degrades 52.3% and 60%, in the same cases. The scalable performance achieved in TMAC demonstrates the effectiveness of the token mechanism in controlling the contention intensity as user population expands. Moveover, TMAC$_{ASM}$ consistently outperforms TMAC$_{OAR}$ by 21% at 54Mbps data rate, and 42.8% at 216Mbps data rate, which reveals the advantage of ASM in supporting high-speed physical layer.

**Average delay:** Table 4.4 lists the average delay of three protocols, DCF, TMAC$_{DCF}$, and TMAC$_{ASM}$ in the simulation scenario identical to the one used in Figure 4.13(b). The table shows that the average delay in TMAC increases much slower than that in DCF, as the user population grows. In specific, the average delay in DCF increases from 0.165$s$ to 5.71$s$ as the number of stations increases form 15 to 285. TMAC$_{DCF}$, adopting token mechanism in the higher tier, reduces the average by up to 39%, while TMAC$_{ASM}$ achieves approximately 70% average delay reduction over various population sizes. The results demonstrate that the token mechanism can efficiently allocate channel share among a large number of stations, thus reducing the average delay. Moveover, ASM improves channel efficiency and further decreases the average delay.

#### 4.2.4.2 Scaling to Different Physical-layer Rates

Within the scenario of 15 contending stations, Figure 4.14 depicts the network throughput obtained by DCF, OAR, and TMAC with the different settings

in the lower tier, as the physical-layer rate varies from 6Mbps to 216Mbps. Note that TMAC$_{ASM}$, with $T_f$ set to $1ms$ and $2ms$, achieves up to 20% and 42% throughput improvement over OAR, respectively. This reveals that TMAC effectively can control protocol overhead at MAC layer especially within the high-capacity physical layer. Our study further reveals that, the overhead incurred by the physical-layer preamble and header is the limiting factor for further improving the throughput achieved by TMAC.

#### 4.2.4.3 Interacting with TCP

In this experiment, we examine the throughput scalability and the fair sharing feature in TMAC when stations, exploiting the rate of 54Mbps, carry out a large file transfer using TCP Reno. The sharing feature is measured by Jain's fairness index [53], which defined as: $(\sum_{i=1}^{n} x_i)^2/(n\sum_{i=1}^{n} x_i^2)$. For station $i$ using the rate of $r_i$,

$$x_i = S_i * T_f/(r_i * T_{ASM}(r_i)),$$

where $S_i$ is the throughput of station $i$. Figure 4.15 plots the network throughput and labels the fairness index obtained with DCF, OAR and TMAC$_{ASM}$ in various user sizes. TMAC demonstrates scalable performance working with TCP. Note that both OAR and DCF experience less than 10% throughput degradation in this case. However, as indicated by the fairness index, both protocols lead to severe unfairness in channel sharing among FTP flows as user size grows. Such unfairness occurs because in DCF and OAR, more than 50% of FTP flows experience service starvation during the simulation run, and 10% flows contribute to more than 90% of the network throughput, as the number of users grows over 75. On the other hand, TMAC, employing the token mechanism, preserves the fair sharing feature while attaining scalable throughput performance at various user sizes.

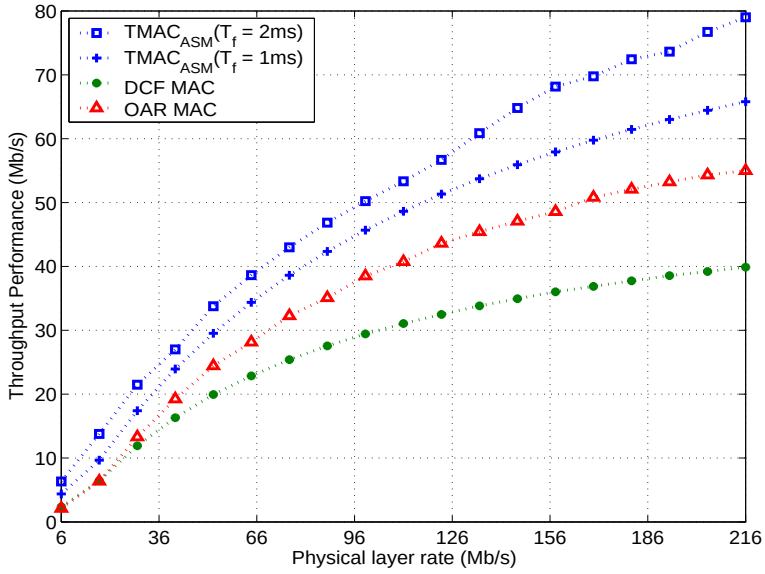


Figure 4.14: Network throughput vs. physical-layer data rates

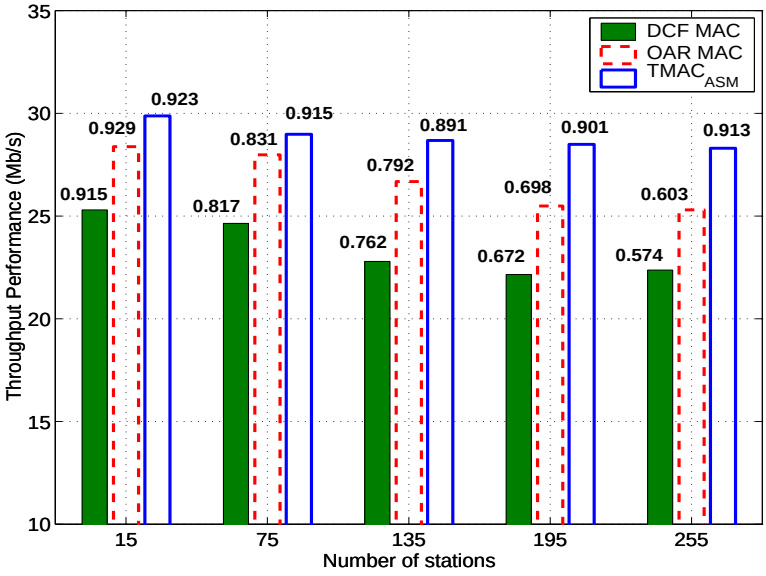


Figure 4.15: Network throughput in TCP experiments

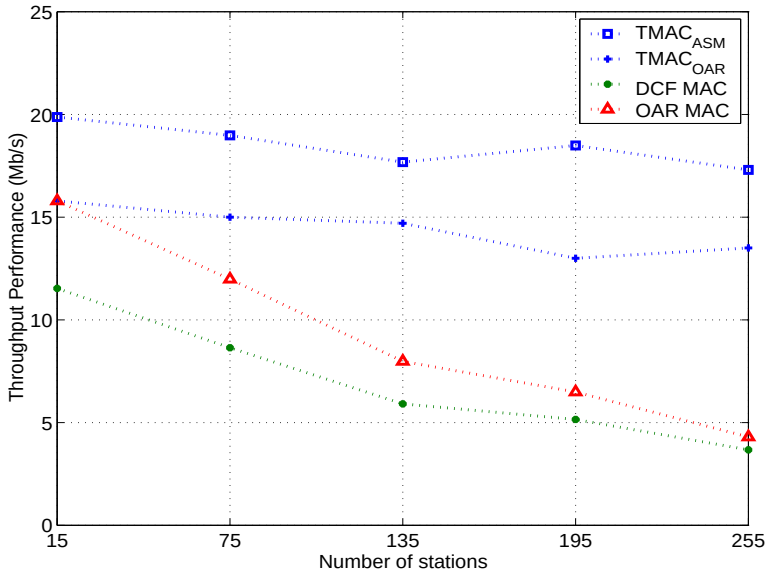


Figure 4.16: Network throughput in Ricean fading channel

#### 4.2.4.4 Ricean fading channel

We now vary channel fading model and study its effects on TMAC with the physical layer specified by 802.11a. Ricean fading channel is adopted in the experiment with $K = 2$, where $K$ is the ratio between the deterministic signal power and the variance of the multipath factor [84]. Stations are distributed uniformly over 400mx400m territory (AP is in the center) and move at the speed of 2.5m/s. The parameter $R_f$ is set at rate of $18Mbps$. Figure 4.16 shows the network throughput of different MAC schemes. These results again demonstrate the scalable throughput achieved by TMAC$_{ASM}$ and TMAC$_{OAR}$ as the number of users grows. TMAC$_{ASM}$ consistently outperforms TMAC$_{OAR}$ by 32% by offering adaptive service share to stations in dynamic channel conditions. In contrast, OAR and DCF experience 72.7% and 68% throughput reduction, respectively, as the user population increases from 15 to 255.

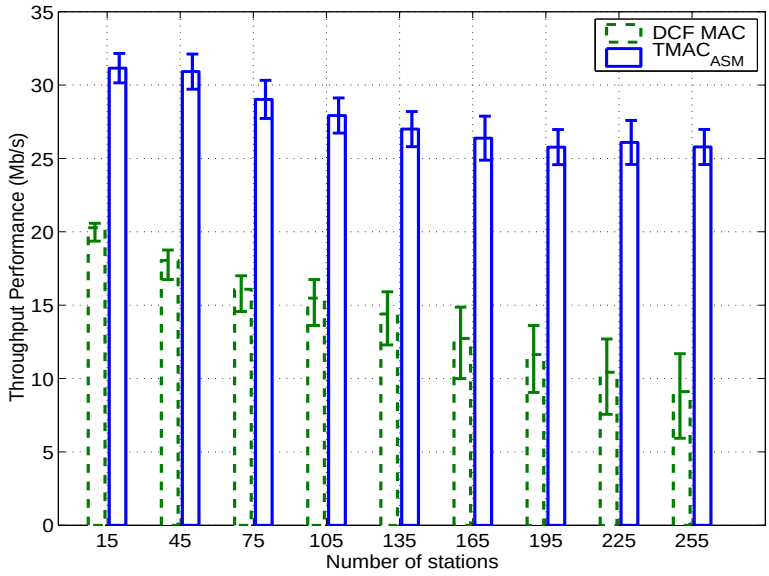


Figure 4.17: Network throughput vs. the number of stations

#### 4.2.4.5 Active station variation and token losses

We examine the effect of variations in the number of active stations caused and of token losses. During the 100-second simulation, 50% stations periodically enter 10-second sleep mode after 10-second transmission. Receiving errors are manually introduced, which causes loss of the token message in nearly 20% of active stations. The average of network throughput in TMAC and DCF is plotted in Figure 4.17 and the error bar shows the maximum and the minimum throughput observed in 10-second interval. When the user size increases from 15 to 255, DCF suffers from throughput reduction up to approximately 55%. It also experiences large variation in the short-term network throughput, indicated by the error bar. In contrast, TMAC achieves stable performance and scalability in the network throughput, despite the fact that the throughput degrade by up to 18% in the same case. Several factors that contribute to the throughput reduction in TMAC include the wastage of TSP, the overhead of membership management and the cost of token loss.

Table 4.5: Throughput (Mbps) and fairness index

| MAC type | 802.11MAC | TMAC ($R_f$ = 216 Mbps) | TMAC ($R_f$ = 108 Mbps) | TMAC ($R_f$ = 54 Mbps) | TMAC ($R_f$ = 24 Mb |
|---|---|---|---|---|---|
| 24Mbps flows | 6.649 | 4.251 | 1.922 | 1.198 | 0.910 |
| 54Mbps flows | 6.544 | 8.572 | 11.282 | 5.004 | 4.695 |
| 108Mbps flows | 6.655 | 12.660 | 15.489 | 20.933 | 10.649 |
| 216M/bs flows | 6.542 | 17.795 | 20.986 | 28.811 | 45.136 |
| All flows | 26.490 | 43.278 | 49.679 | 55.946 | 61.390 |
| Fairness Index | 0.6246 | 0.9297 | 0.9341 | 0.9692 | 0.9372 |

Table 4.6: Network throughput (Mbps) vs. $T_f$ and $A_f$

| $T_f$ | $0ms$ | $1ms$ | $2ms$ | $3ms$ | $4ms$ | $5ms$ |
|---|---|---|---|---|---|---|
| 54Mbps | 20.40 | 25.33 | 28.91 | 32.10 | 32.93 | 33.40 |
| 216Mbps | 35.16 | 70.70 | 76.19 | 78.35 | 79.31 | 79.88 |
| $A_f$ | 1 | 2 | 3 | 4 | 5 | 6 |
| 216Mbps | 78.35 | 93.92 | 95.91 | 97.29 | 98.94 | 101.72 |

#### 4.2.4.6 Design parameters $A_f$ and $T_f$

We now evaluate the impacts of the design parameters $T_f$ and $A_f$. We adopt scenarios similar to the case $A$, and fix the number of users as 50. The reference transmission duration $T_f$ varies with $A_f$ set to 1, where $T_f$ of $0ms$ grants one packet transmission as in the legacy MAC. Next, to quantify the effect of the block ACK size, we tune $A_f$ from 1 to 6, with $3ms$ $T_f$.

Table 4.6 presents the network throughput obtained with TMAC as the design parameters of $T_f$ and $A_f$ vary. When $T_f$ changes from $0ms$ to $5ms$, the aggregate throughput improves by 63.7% at 54Mbps data rate, and 127% with 216Mbps rate. Tuning the parameter $A_f$ can further improve the throughput to more than 100Mbps. The improvements show that the overhead caused by per-packet contention and acknowledgement has been effectively reduced in TMAC.

#### 4.2.4.7 Exploiting rate diversity

In the final set of experiments, we demonstrate that TMAC can adaptively leverage multirate capability at each station to further improve the aggregate throughput. We use the fairness index defined in Section 4.2.4.3. We consider the simulation setting of eight stations in one token group. Each station carries a UDP flow to fully saturate the network. There are four transmission rate options, 24Mbps, 54Mbps, 108Mbps and 216Mbps. Each pair of stations randomly chooses one of the four rates. The results are obtained from averaging over 5 simulation runs.

Table 4.5 enumerates the aggregate throughput and the fairness index for flows transmitting at the same rate, using the 802.11 MAC, and TMAC with different $R_f$ settings. TMAC enables high-rate stations to increasingly exploit their good channel conditions by granting the high-rate nodes more time share than the low-rate stations. This is realized by reducing a single parameter $R_f$. TMAC acquires 65%, 87%, 111% and 133% overall throughput gains compared with the legacy MAC as adjusting $R_f$ to 216Mbps, 108Mbps, 54Mbps and 24Mbps, respectively.

Moreover, The fairness index for TMAC design is close to 1 in every case, which indicates the effectiveness of the adaptive sharing scheme. The fairness index of DCF MAC is 0.624 in temporal units. DCF results in such a severe bias because it neglects the heterogeneity in channel quality experienced by stations and offers them equal throughput share. In summary, by lowering the access priority of low-rate stations that are nevertheless not in good channel conditions, TMAC provides more transmission opportunities for high-rate stations perceiving good channels. This feature is important for high-speed wireless LANs and mesh networks to mitigate the severe aggregate throughput degradation incurred by low-rate stations. The lower channel sharing portion by a low-rate station also moti-

vates it to move to a better spot or to improve its reception quality. In either case, the system throughput is improved.

### 4.2.5 Discussions

In this section, we first discuss alternative designs to address the scaling issues in high-speed wireless LANs. We then present a few issues relevant to TMAC. We will discuss the prior work related to TMAC in detail in Section 4.2.6.

TMAC employs a centralized solution to improve user experiences and provide three scaling properties, namely user population scaling, physical-layer capacity scaling, and protocol overhead scaling. The design parameters used in TMAC can be customized for various scenarios, which is especially useful for wireless Internet Service Providers (ISP) to improve the service quality. One alternative scheme for supporting large user sizes is to use the distributed method to tune CW. Such a method enables each node to estimate the perceived contention level and thereafter choose the suitable CW. (e.g., AOB [57], Idle Sense [61]). Specifically, the slot utilization, or the number of idle time slots is measured and serves as the input to derive CW in DCF MAC.

The distributed scheme for adjusting CW will have difficulty in providing scaling performance especially in high-speed wireless LANs. First, the distributed scheme derives CW by modeling the DCF MAC. The result can not be readily applied to high-speed wireless networks. The MAC design in high-speed wireless networks, such as IEEE 802.11n, largely adopts the existing schemes proposed in IEEE 802.11e [43]. In 802.11e, several access categories are defined to offer differentiated services in supporting various applications. Each access category uses different settings of the deferring time period, CW, and the transmission duration. The new MAC protocol inevitably poses challenges to the distributed schemes based on modeling DCF, the simpler version of 802.11e. Second,

the distributed scheme mainly considers tuning CW to match the contention intensity. The scaling issues of protocol overhead and physical-layer capacity are not explicitly addressed. Moreover, the distributed scheme requires each node to constantly measure contention condition for adjusting CW. The design incurs extra management complexity at APs due to lack of the control of user behaviors.

The problem with the distributed scheme of CW tuning may be solvable in high-speed wireless LANs, but it is clear that a straightforward approach using the centralized control prevents several difficulties. TMAC can support access categories by announcing the corresponding parameters, such as CW, in the token messages. More sophisticated schedulers (e.g., weighted round robin) can be adopted to arrange the token groups in order to meet the quality-of-service (QoS) requirements of various applications. The adaptive service model enables packet aggregation and differentiates the time share allocated to the high-rate and low-rate stations to leverage data-rate diversities. In addition, most computation complexity in TMAC occurs at APs, while client devices only require minor changes to handle received tokens. The design parameters offers wireless ISPs extra flexibility to control system performance and fairness model. The two-tier design adopted by TMAC extracts benefits of the random access and the polling mechanism, hence provides a highly adaptable solution for the next generation, high-speed wireless LANs.

We now discuss several issues relevant to the TMAC design.

**Backward Compatibility:** TMAC so far mainly focuses on operating in the infrastructure mode. Since the fine-grained channel access is still based on CSMA/CA, TMAC can coexist with stations using the current 802.11 MAC. AP still uses the token distribution and reference parameter set to coordinate channel access among stations supporting TMAC. However, the overall MAC performance will degrade as a larger number of regular stations contend for channel

access.

**Handling Misbehaving Stations:** Misbehaving stations expose them by acquiring more channel share than its fair share during its batch transmission or contending for channel access when it does not possess the current TGID. We can mitigate these misbehaving stations by monitoring and policing them via central AP. Specifically, the AP can keep track the channel time each station received, and calculates its fair share based on the collected information of the station transmission rate and other reference parameter settings. When the AP detects an overly aggressive station, say, access time beyond certain threshold, it temporarily revokes channel access right of the station. This can be realized via the re-authentication mechanism provided by the current 802.11 MAC management plane.

**Power Saving:** TMAC supports power saving and also works with the power saving mechanism (PSM) defined in 802.11. In TMAC, time is divided into token service periods, and every node in the network is synchronized by periodic token transmissions. So every node will wake up at beginning each token service period at about the same time to receive token messages. The node that does not belong to the current token group can save energy by going into doze mode. In doze mode, a node consumes much less energy compared to normal mode, but cannot send or receive packets. Within the token service period, PSM can be applied to allow a node to enter the doze mode only when there is no need for exchanging data in the prevailing token period.

### 4.2.6 Related Work

A number of well-known contention-based channel access schemes have been proposed in literature, starting from the early ALOHA and slotted ALOHA protocols [55], to the more recent 802.11 DCF [75], MACA [77], MACAW [91].

These proposals, however, all face the fundamental problem that their throughput drops to almost zero as the channel load increases beyond certain critical point [89]. This issue leads to the first theoretical study of network performance as the user population varies [89]. The study further stimulates the recent work [9, 29,30,57,61,96] on dynamically tuning the backoff procedure to reduce excessive collisions within large user populations. However, backoff tuning generally requires detailed knowledge of the network topology and traffic demand, which are not readily available in practice. TMAC differs from the above work in that it addresses the scalability issues in a two-tier framework. The framework incorporates a higher-tier channel regulation on top of the contention-based access method to gracefully allocate channel resource within different user populations. In the meantime, TMAC offers capacity and protocol overhead scalability through an adaptive sharing model. Collectively, TMAC controls the maximum intensity of resource contention and delivers scalable throughput for various user sizes with minimal overhead.

A number of enhanced schemes for DCF have been proposed to improve its throughput fairness model [75] in wireless LANs. Equal temporal share model [10, 16] and throughput propositional share model [26] generally grant each node the same share in terms of channel time to improve the network throughput. In 802.11e and 802.11n, access categories are introduced to provide applications different priorities in using the wireless medium. In TMAC, the existing models can be applied to the lower-tier design directly. To offer the flexibility of switching the service model, we exploit the adaptive service model, which allows administrators to adjust the time share for each station based on both user demands and the perceived channel quality. To further reduce the protocol overhead, TMAC renovates the block ACK technique proposed in 802.11e [43] by removing the tedious setup and tear-down procedures, and introduces an adjustable parameter

for controlling the block size. More importantly, TMAC is designed for a different goal - it is to tackle the three scalability issues in next-generation wireless data networks.

Reservation-based channel access methods typically exploit the polling model [59, 88] and dynamic TDMA schemes in granting channel access right to each station. IBM Token Ring [5] adopts the polling model in the context of wired network by allowing a token to circulate around the ring network. Its counterpart in wireless network includes PCF [75] and its variants [43]. The solutions of HiperLAN/2 [4] and [39] are based on dynamic TDMA and transmit packets within the reserved time slots. All these proposals use reservation-based mechanisms in fine-time-interval channel access for each individual station. In contrast, the polling model applied in TMAC achieves coarse-grained resource allocation for a group of stations to multiplex bursty traffic loads for efficient channel usage.

Some recent work has addressed certain aspect of the scalable MAC design. The work by [82] recognized the impact of scalability in MAC protocol design, but did not provide concrete solutions. Commercial products [67, 81] have appeared in the market that claimed scalable throughput in the presence of about 30 users for their 802.11b APs. ADCA [97], our previous work, is proposed to reduce the protocol overhead as the physical-layer rate increases. The method of tuning CW based on idle slots [61] have been explored to manage channel resource and fairness for large user sizes. Multiple-channel [50] and cognitive radios [19] offer the promise of spectrum agility to increase the available resources by trading off the hardware complexity and cost. Inserting an overlay layer [12] or using multiple MAC layers [11,15] has been exploited to increase network efficiency. However, an effective MAC framework that is able to tackle all three key scalability issues has not yet been adequately addressed.

## 4.3 Chapter 4 in Summary

Today WLANs are going through similar development and deployment cycles that wired Ethernet has been through in the past three decades - expanding deployment in more diversified environments, driving the speed to orders of magnitude higher, and keeping low protocol overhead. To support the continual growth of WLANs, we study user behaviors based on the recent measurements of WLANs deployed in real world. Our study reveals that the fixed-width channelization technique and the 802.11 MAC design used in WLANs are inherently incapable of efficiently coping with the spatially non-uniform and temporally dynamic user demand, which is prevalent in most infrastructure networks deployed today.

In this chapter, we have demonstrated that by moving beyond these predetermined channels of fixed width, a significant increase of both system capacity and per-client fairness can be achieved. Made feasible by recent advances in hardware technology, we propose a system and a set of algorithms that efficiently and dynamically allocate center-frequencies and channel widths to APs as a function of their traffic loads.

Furthermore, we propose a new scalable MAC solution, which takes a two-tier design approach, employing *centralized, coarse-grained channel regulation* at the higher tier, and *distributed, fine-grained random access* in the lower tier. The higher tier organizes stations into multiple token groups and permits only the stations in one group to contend for channel access at a time. This token mechanism effectively controls the maximum intensity of channel contention and gracefully scales to diverse population sizes. At the lower tier, we propose an adaptive channel sharing model working with the distributed random access, which largely reduces protocol overhead and exploits rate diversity among stations. The extensive analysis and simulations have confirmed effectiveness of the dynamic

channelization structure and the scalable MAC design.

# Chapter 5

# Conclusion

*Simplicity is the ultimate sophistication.*

*— Leonardo da Vinci*

*Make everything as simple as possible,*

*but not simpler.*

*— Albert Einstein*

As the key to the provision of wireless communication and information services, the radio frequency spectrum is a naturally limited resource of extraordinary value. In recent years, the "artificial" spectral shortage problem, caused by the strict control of government agents, has greatly encumbered advances of innovative wireless systems. This dissertation is focused on tackling this problem by improving the spectrum utilization and efficiency. The proposed solution is based upon the novel idea of dynamic spectrum allocation, which assigns a wireless node with an amount of spectrum as a function of total bandwidth of available spectrum, the number of contending nodes, and traffic load distribution in a local network environment. Without requiring any prior knowledge about the available spectrum, our proposed allocation scheme adaptively creates the appropriate number of channels to maximize parallel transmissions, and adjusts

channel-width assigned to each node for balancing traffic load variations across a network. Such a spectrum allocation method can make use of dynamic white spaces, maximize parallelism for high spectrum efficiency and reduce interferences by separating communicating nodes into different frequency bands. We have proposed two parts of the solution including the KNOWS system and the dynamic channelization structure for WLANs. Through analysis and extensive simulations, we have demonstrated and quantified gains of network throughput and fairness achieved by the dynamic spectrum allocation as compared with the IEEE 802.11 networks that are based on the fixed spectrum allocation paradigm.

As the first part of our solution, we propose KNOWS, which is a system encompassing new hardware, an enhanced MAC protocol and a spectrum allocation algorithm for efficiently utilizing unused portions of the licensed spectrum to perform unlicensed operations. KNOWS cooperatively detects incumbent operators and efficiently shares the vacant spectrum among unlicensed users. We encapsulate the proposed dynamic spectrum allocation concept in a series of time-spectrum blocks, which are units of spectrum allocations. A time-spectrum block specifies which node should use how wide a spectrum-band at which center-frequency and for how long. We then propose and evaluate a distributed solution, called b-SMART, which enables nodes to use time-time blocks for allocating the white spaces in a fine timescale. We have shown that KNOWS significantly increases the network capacity compared to IEEE 802.11-based systems through analysis and simulation studies.

We improve efficiency of WLANs using the dynamic spectrum allocation as the second part of our solution. We implement the dynamic allocation concept by proposing a dynamic channelization structure for WLANs. By dynamically allocating variable-width channels, the dynamic channelization is able to cope with variations of user demands in WLANs. If few APs are in the WLAN, the struc-

ture creates channels with larger widths, enabling the clients to communicate at a higher speed. This new channelization provides better fairness than IEEE 802.11 because heavily-loaded APs get a larger spectrum to balance the per-client throughput across the network. We also propose a scalable MAC design to support the dynamic channelization. The new MAC design is scalable to variations of clients who operate in the same channel and compete for channel access.

The efficient management of the radio spectrum has attracted many proposals which could be termed "dynamic spectrum allocation". We have particularly focused on dynamic allocations of white spaces in the TV bands and the unlicensed spectrum. Our solutions are compliant with the policy of "spectrum commons" adopted by government agents, such as the FCC, for improving spectrum efficiency. The other promising approach for solving the spectral shortage problem is to enable a market-based approach, which allows licensees to temporarily lease or transfer the spectrum unused by primary users to secondary users. In 2003, the FCC adopted the its first Report and Order and Further Notice of Proposed Rulemaking (FCC 03-113), in which it established new policies and procedures to facilitate secondary market of spectrum through spectrum leasing, and specified procedures for approving license assignments and transfers of control. In 2004, the FCC adopted the Second Report and Order, Order on Reconsideration, and Second Further Notice (FCC 04-167), in which it provided for immediate processing of certain qualifying spectrum leasing, license assignment and transfer transactions. The market-based approach is superior to the conventional "command and control" policy because they allow change and competition, and move spectrum to their highest valued uses in society. In order to enable the market-based solution, we need to tackle a number of challenges, including network architecture design, spectrum allocation, secondary user authentication, spectrum usage authorization, spectrum usage control, billing and etc. Through

this research experience we believe that the dynamic spectrum allocation concept proposed in this dissertation can also be applied to reduce interferences and maximize spectrum efficiency and revenue in the secondary market of the spectrum. This dissertation does not intend to provide a comprehensive solution to enable the secondary market; rather, it reveals great advantages of the proposed dynamic spectrum allocation through rigid research and lays down a solid foundation for future exploration on this topic.

# Bibliography


[1] . Aruba Networks, http://www.arubanetworks.com.

[2] . Atheros Communications, http//www.atheros.com/pt/whitepapers/atheros-superg-whitepaper.pdf.

[3] . Cisco Aironet Adapter. http://www.cisco.com/en/US/products/hw/wireless/ps4555/products_data_sheet09186a00801ebc29.html.

[4] . Hiperlan/2 EN 300 652 V1.2.1(1998-07), Function Specification,ETSI.

[5] . IEEE 802.5: Defines the MAC layer for Token-Ring Networks.

[6] . IEEE 802.11n: Wireless LAN MAC and PHY Specifications: Enhancements for Higher Throughput, 2005.

[7] . IEEE802.11n: Sync Proposal Technical Specification, doc. IEEE 802.11-04/0889r6, May 2005.

[8] G. Holland, N. Vaidya, and P. Bahl. A Rate-Adaptive MAC Protocol for Multi-Hop Wireless Networks. In *ACM MobiCom*, 2001.

[9] G. Tan and J. Guttag. Improving Protocol Capacity with Model-Based Frame Scheduling in IEEE 802.11-Operated WLANs. In *ACM MobiCom*, 2003.

[10] G. Tan and J. Guttag. Time-based Fairness Improves Performance in Multi-rate Wireless LANs. In *ACM USENIX*, 2004.

[11] A. Farag, A. D. Myers, V. R. Syrotiuk, and G. V. Zruba. Meta-MAC Protocols: Automatic Combination of MAC Protocols to Optimize Performance for Unknown Conditions. *IEEE Journal on Selected Area in Communications*, 18(9):1670–1681, 2000.

[12] A. Rao, I. Stoica. An overlay MAC layer for 802.11 networks. In *ACM MobiSys*, 2005.

[13] N. Ahmed and S. Keshav. SMARTA: A Self-Managing Architecture for Thin Access Points. In *ACM CoNext*, 2006.

[14] A. Mishra V. Brik S. Banerjee A.Srinivasan and W. Arbaugh. Client-driven Channel Management for Wireless LANs. In *IEEE INFOCOM*, 2006.

[15] B. A. Sharp, E. A. Grindrond, and D. Camm. Hybrid TDMA/CSMA Protocol For Self Managing Packet Radio Networks. In *IEEE ICUPC*, 1995.

[16] B. Sadeghi, V. Kanodia, A. Sabharwal, and E. Knightly. Opportunistic Media Access for Multirate Ad Hoc Networks. In *ACM MobiCom*, 2002.

[17] P. Bahl, M. T. Hajiaghayi, K. Jain, V. Mirrokni, L. Qiu, and A. Seberi. Cell Breathing in Wireless LANs: Algorithms and Evaluation. *IEEE Transactions on Mobile Computing*, 2006.

[18] M. Balazinska and P. Castro. Characterizing mobility and network usage in a corporate wireless local-area network. In *ACM MobiSys*, 2003.

[19] C. Doerr, M. Neufeld, J. Fifield, T. Weingart, D.C. Sicker, D. Grunwald. MultiMAC - An Adaptive MAC Framework for Dynamic Radio Networking. In *IEEE DySPAN*, 2005.

[20] Ranveer Chandra, Jitendra Padhye, Alec Wolman, and Brian Zill. A Location-Based Management System for Enterprise Wireless LANs. In *Fourth Symposium on Networked Systems Design and Implementation (NSDI)*, April 2007.

[21] Comments of the Consumer Electronics Association. FCC Media Bureau Docket 04-210, Inquiry Into Over-the-Air Broadcast Television Viewers, August 2004, p.2.

[22] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. John Wiley and Sons, Inc., 1991.

[23] D. Kotz and K. Essien. Analysis of a Campus-wide Wireless Network. In *ACM MobiCom*, 2002.

[24] D. Scaperoth, B. Le, T. Rondeau, D. Maldonado, C. W. Bostian, and S. Harrison. Cognitive Radio Platform Development for Interoperability. In *IEEE MILCOM*, 2006.

[25] D. Tang and M. Baker. Analyis of a Local-Area Wireless Network. In *ACM MobiCom*, August 2000.

[26] D. Tse. Multiuser Diversity in Wireless Networks: Smart Scheduling, Dumb Antennas and Epidemic Communication. In *IMA Wireless Workshop*, 2001.

[27] R. Jain D.M. Chiu. Analysis of the increase and decrease algorithms for congestion advoidance in computer networks. *Computer Networks and ISDN Systems*, Jun. 1989.

[28] Ed Thomas. DTV 201: How the DTV Transition Can Move the Nation from Broadcast to Broadband, New America Foundation and House Fu-

ture of American Media Caucus Luncheon, September 7, 2005. Available at: http://www.newamerica.net/Download_Docs/pdfs/Doc_File_2547_1.pdf.

[29] F. Cali, M. Conti, and E. Gregori. IEEE 802.11 Protocol: Design and Performance Evaluation of an Adaptive Backoff Mechanism. *IEEE Journal on Selected Area in Communications*, 2000.

[30] F. Cali, M. Conti, and E. Gregori. Dynamic Tuning of the IEEE 802.11 Protocol to Achieve a Theoretical Throughput Limit. *IEEE Transactions on Networking*, 2002.

[31] FCC. Unlicensed Operation in the TV Broadcast Bands Notice of Proposed Rulemaking (NPRM), ET Docket No. 04-186, May 2004.

[32] FCC Annual Assessment of the Status of Competition in the Market for the Delivery of Video Programming. Eleventh Annual Report (February 2005), Table B-1 and Sixth Annual Report (January 2000), Table C-1. Household OTA penetration is calculated as the percentage of households without MVPD service.

[33] Federal Communications Commission Spectrum Policy Task Force. FCC Report of the Spectrum Efficiency Working Group, November 2002. http://www.fcc.gov/sptf/files/SEWGFinalReport_1.pdf.

[34] G. Bianchi. Performance Analysis of the IEEE 802.11 Distributed Coordination Function. *IEEE Journal on Selected Area in Communications*, 18(3):535–547, 2000.

[35] Naveen Garg and Jochen Koenemann. Faster and Simpler Algorithms for Multicommodity Flow and other Fractional Packing Problems. In *Proc. of the $39^{th}$ Annual Symposium on Foundations of Computer Science (FOCS)*, 1998.

[36] GNU Software Radio. http://www.gnu.org/software/gnuradio/.

[37] A. Graf, M. Stumpf, and G. Weissenfels. On Coloring Unit Disk Graphs. *Algorithmica*, 20, 1998.

[38] Piyush Gupta and P. R. Kumar. The Capacity of Wireless Networks. *IEEE Transactions on Information Theory*, IT-46(2):388–404, 2000.

[39] I. Cidon and M. Sidi. Polling Systems: Applications, Modeling, and Optimization. *IEEE Transactions on Computers*, 38(10):1353–1361, Oct. 1989.

[40] IEEE 802.11b/D3.0. Wireless LAN Medium Access Control(MAC) and Physical (PHY) Layer Specification: High Speed Physical Layer Extensions in the 2.4 GHz Band, 1999.

[41] IEEE 802.22 WRAN WG. www.ieee802.org/22/.

[42] IEEE Std 802.11a-1999 Part 11: . Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY).

[43] IEEE Std 802.11e/D8.0 Part 11: . Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY).

[44] J. Elson and D. Estrin. Time Synchronization for Wireless Sensor Networks. In *IEEE IPDPS*, 2001.

[45] J. Hao and L. Perrier. CTabu Search for the Frequency Assignment Problem in Cellular Radio Networks. In *French Workshop on Practical Solving of NP-Complete Problems, Dijon, France*, March 2005.

[46] J. Hao and R. Drone. Study of Genetic Search for the Frequency Assignment Problem. In *Artificial Evolution, Lecture Notes in Computer Science*, volume 42, 19966.

[47] J. Mitola III . Cognitive Radio: an Integrated Agent Architecture for Software Defined Radio. In *Ph.D Thesis, KTH Royal Institute of Technology, Stockholm, Sweden*, 2000.

[48] J. Padhye, S. Agarwal, V. Padmanabhan, L. Qiu, A. Rao and B. Zill. Estimation of Link Interference in Static Multi-hop Wireless Networks. In *In Proc. of Internet Measurement Conference (IMC)*, 2005.

[49] J. Proakis. *Digital Communications*. McGraw Hill, 2001.

[50] N. Vaidya J. So. A Multi-channel MAC Protocol for Ad Hoc Wireless Networks. Technical report, Dept. of Electrical and Computer Engeneering, University of Illinois, USA, Jan. 2003.

[51] Ravi Jain, Dan Lelescu, and Mahadevan Balakrishnan. Model T: An Empirical Model for User Registration Patterns in a Campus Wireless LAN. In *ACM MobiCom*, pages 170–184, 2005.

[52] J.H. Snider. Reclaiming the Vast Wasteland THE ECONOMIC CASE FOR REALLOCATING THE UNUSED SPECTRUM (WHITE SPACE) BETWEEN TV CHANNELS 2 AND 51 TO UNLICENSED SERVICE.

[53] K. Jain, J. Padhye, V. Padmanabhan, and L. Qiu. Impact of Interference on Multi-hop Wireless Network Performance. In *ACM MobiCom*, 2003.

[54] K. Romer. Time Synchronization in Ad Hoc Networks. In *ACM MobiHoc*, 2001.

[55] L. Kleinrock and F. A. Tobagi. Packet Switching in Radio Channels: Part I – Carrier Sense Multiple-Access Modes. *IEEE Trans. on Wireless Communications*, 1975.

[56] P. Kyasanur, J. Padhye, and P. Bahl. On the Efficacy of Separating Control and Data Into Different Frequency Bands. In *BROADNETS*, pages 646–655, 2005.

[57] L. Bononi, M. Conti, and E. Gregori. Runtime Optimization of IEEE 802.11 Wireless LANs Performance. *IEEE Transactions on Parallel and Distributed Systems*, 15(1):66–80, Jan. 2004.

[58] L. Narayanan. Channel assignment and graph multicoloring. In *Handbook of wireless networks and mobile computing*, 2002.

[59] H. Levy and M. Sidi. Polling Systems: Applications, Modeling, and Optimization. *IEEE Transactions on Communications*, Oct. 1990.

[60] M. Duque-Anton, D. Kunz, and B. Ruber. Channel assignment for Cellular Radio using Simulated Annealing. *IEEE Trans. on Vehicular Technology*, 42, 1993.

[61] M. Heusse, F. Rousseau, R. Guillier, and A. Duda . Idle Sense: An Optimal Access Method for High Throughput and Fairness in Rate Diverse Wireless LANs. In *IEEE SIGCOMM*, 2005.

[62] M. M. Buddhikot, P. Kolodzy, S. Miller, K. Ryan, and J. Evans . DDIMSUM-Net: New directions in wireless networking using coordinated dynamic spectrum access. In *IEEE WoWMoM05*, June 2005.

[63] L. Ma, X. Han, and C. Shen. Dynamic Open Spectrum Sharing MAC Protocol for Wireless Ad Hoc Networks. In *IEEE Dyspan*, 2005.

[64] R. Maheshwari, H. Gupta, and S. Das. MAC Protocol for Multiple Channels. In *IEEE SECON*, 2006.

[65] M.V. Marathe, H. Breu, H.B. Hunt III, S. S. Ravi, and D.J. Rosenkrantz. Simple Heuristics for Unit Disk Graphs. *Networks*, 25, 1995.

[66] D. W. Matula and L. L. Beck. Smallest-last ordering and clustering in graph coloring algorithms. *Journal of the ACM*, 30(3):417–427, 1983.

[67] Merunet. http://www.computerworld.com/mobiletopics/mobile/story/0,10801,658

[68] V. Mhatre and Konstantina Papagiannaki. Optimal Design of High Density 802.11 WLANs. In *ACM CoNext*, 2006.

[69] Arunesh Mishra, Vivek Shrivastava, Suman Banerjee, and William Arbaugh. Partially Overlapped Channels not Considered Harmful. *SIGMETRICS Perform. Eval. Rev.*, 34(1):63–74, 2006.

[70] T. Moscibroda, R. Wattenhofer, and Y. Weber. Protocol Design Beyond Graph-Based Models. In *ACM HotNets*, 2006.

[71] T. Moscibroda, R. Wattenhofer, and Y. Weber. Protocol Design Beyond Graph-Based Models. In *ACM HotNets*, 2006.

[72] N. Funabiki and Y. Takefuji. A Neural Network Parallel Algorithm for Channel Assignment Problems in Cellular Radio Networks. *IEEE Trans. on Vehicular Technology*, 41, 1998.

[73] OECD Communications Outlook 2005. (Paris, France: OECD Publishing, 2005), p. 222.

[74] Office of Engineering and Technology. Projected Schedule for Proceeding on Unlicensed Operation in the TV Broadcast Bands, 2006.

[75] B. O'Hara and A. Petrick. *IEEE 802.11 Handbook, A Designer's Companion*. IEEE Press, 1999.

[76] P. Bahl, R. Chandra, and J. Dunagan. SSCH: Slotted Seeded Channel Hopping for Capacity Improvement in IEEE 802.11 Ad-Hoc Wireless Networks. In *ACM MobiCom*, 2004.

[77] P. Karn. MACA: A New Channel Access Method for Packet Radio. In *The 9th Computer Networking Conference*, 1999.

[78] P. Kyasanur, J. So, C. Chereddi, and N. H. Vaidya. Multi-Channel Mesh Networks: Challenges and Protocols. In *IEEE Wireless Communications*, April 2006.

[79] C. Perkins, E. Belding-Royer, and S. Das. Ad hoc On-Demand Distance Vector (AODV) Routing. In *IETF RFC 3561*, July 2003.

[80] Ping Ding, JoAnne Holliday, Aslihan Celik. Dynamic Scheduling of PCF Traffic in an Unstable Wireless LAN. In *Proceedings of IEEE Consumer Communications and Networking Conference*, 2005.

[81] QualNet. http://www.qualnet.com/.

[82] R. Karrer, A. Sabharwal, and E. Knightly. Enabling Large-Scale Wireless Broadband: The Case for TAPs. In *IEEE HotNet*, 2003.

[83] R. Mukhopadhyay, Y. Park, P. Sen, N. Srirattana, J. Lee, C. Lee, S. Nuttinck, A. Joseph, J.D. Cressler, J. Laskar. Reconfigurable RFICs in Silicon-based technologies for a compact intelligent RF front-end. *IEEE Transactions on Microwave Theory and Techniques*,, 53(1):81–93, Jan. 2005.

[84] T.S. Rapport. *Wireless Communications: Principles and Practice*. 2nd Edition, 2005.

[85] Shared Spectrum Company. http://www.sharedspectrum.com.

[86] J. So and N. H. Vaidya. Multi-Channel MAC for Ad Hoc Networks: Handling Multi-Channel Hidden Terminals Using a Single Transceiver. In *ACM MobiHoc*, 2004.

[87] Karthikeyan Sundaresan and Konstantina Papagiannaki. The Need for Cross-layer Information in Access Point Selection Algorithms. In *Internet Measurment Conference*, pages 257–262, 2006.

[88] F. Tobagi and L. Kleinrock. Packet Switching in Radio Channels: Part III – Polling and (Dynamic) Split-Channel Reservation Multiple Access. *IEEE Trans. on Wireless Communications*, 1976.

[89] F. Tobagi and L. Kleinrock. Packet Switching in Radio Channels: Part IV – Stability Considarations and Dynamic Control in Carrier Sense Multiple Access. *IEEE Trans. on Wireless Communications*, 1977.

[90] V. Bharghavan. A Dynamic Addressing Scheme for Wireless Media Access. In *IEEE ICC*, 1995.

[91] V. Bharghavan, A. Demers, S. Shenker, L. Zhang. MACAW: A Media Access Protocol for Wireless LANs. In *ACM SIGCOMM*, 1999.

[92] V. Brik, E. Rozner, S. Banerjee, and P. Bahl. DSAP: A Protocol for Coordinated Spectrum Access. In *IEEE Dyspan*, 2005.

[93] W. Stark. Analysis of Hidden Terminals in Cognitive Radio Network.

[94] 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 Mobile Ad Hoc Networks. In *International Symposium on Parallel Architectures, Algorithms and Networks (I-SPAN)*, 2000.

[95] X. Meng, S. Wong, Y. Yuan, S. Lu. Characterizing Flows in Large Wireless Data Networks. In *ACM MobiCom*, 2004.

[96] Y. Kwon, Y. Fang, and H. Latchman. A Novel MAC Protocol with Fast Collision Resolution for Wireless LANs. In *IEEE INFOCOM*, 2003.

[97] Y. Yuan, D. Gu, W. Arbaugh, J. Zhang. High-Performance MAC for High-Capacity Wireless LANs. In *IEEE ICCCN*, 2004.

[98] Y. Yuan, P. Bahl, R. Chandra, P. A. Chou, J. I. Ferrell, T. Moscibroda, S. Narlanka and Y. Wu. KNOWS: Kognitive Networking Over White Spaces. In *IEEE Dyspan*, 2007.

[99] Y. Yuan, P. Bahl, R. Chandra, T. Moscibroda, S. Narlanka and Y. Wu. Allocating Dynamic Time-Spectrum Blocks for Cognitive Radio Networks. In *ACM MobiHoc*, 2007.

[100] Y. Yuan, P. Bahl, R. Chandra, T. Moscibroda, Y. Wu . Characterizing Flows in Large Wireless Data Networks.

[101] Y. Yuan, W. Arbaugh, S. Lu. Analysis of a Campus-wide Wireless Network. *EURASIP Journal on Wireless Communications and Networking* , 2007.

[102] Z. Ji, Y. Yang, J. Zhou, M. Takai, and R. Bagrodia. Exploiting Medium Access Diversity in Rate Adaptive Wireless LANs. In *ACM MobiCom*, 2004.

[103] J. Zhao, H. Zheng, and G. Yang. Distributed Coordination in Dynamic Spectrum Allocation Networks. In *IEEE Dyspan*, 2005.

[104] Q. Zhao, L. Tong, and A. Swami. Decentralized Cognitive MAC for Dynamic Spectrum Access. In *IEEE Dyspan*, 2005.