ANALYSIS AND MITIGATION OF INTERFERENCE IN MULTI-RADIO MULTI-CHANNEL WIRELESS MESH NETWORKS

A DISSERTATION SUBMITTED TO
THE DEPARTMENT OF COMPUTER ENGINEERING
AND THE GRADUATE SCHOOL OF ENGINEERING AND SCIENCE
OF BILKENT UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

By
Alper Rifat Uluçınar
July, 2013

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philosophy.

Assoc. Prof. Dr. İbrahim Körpeoğlu (Advisor)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philosophy.

Prof. Dr. Ezhan Karaşan

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philosophy.

Prof. Dr. Özgür Ulusoy

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philosophy.

Assoc. Prof. Dr. Uğur Güdükbay

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philosophy.

Prof. Dr. Adnan Yazıcı

Approved for the Graduate School of Engineering and Science:

Prof. Dr. Levent Onural
Director of the Graduate School

ABSTRACT

ANALYSIS AND MITIGATION OF INTERFERENCE IN MULTI-RADIO MULTI-CHANNEL WIRELESS MESH NETWORKS

Alper Rifat Uluçınar
Ph.D. in Computer Engineering
Supervisor: Assoc. Prof. Dr. İbrahim Körpeoğlu
July, 2013

Wireless mesh networking, which is basically forming a backbone network of mesh routers using wireless links, is becoming increasingly popular for a broad range of applications from last-mile broadband access to disaster networking or P2P communications, because of its easy deployment, self-forming, self-configuration, and self-healing properties. The multi-hop nature of wireless mesh networks (WMNs) aggravates inter-flow interference and causes intra-flow interference and severely limits the network capacity. One technique to mitigate interference and increase network capacity is to equip the mesh routers with multiple radios and use multiple channels. The radios of a mesh router can then simultaneously send or receive packets on different wireless channels. However, careful and intelligent radio resource planning, including flow-radio and channel assignment, is necessary to efficiently make use of multiple radios and channels. This first requires analyzing and modeling the nature of co-channel and adjacent channel interference in a WMN.

Through real-world experiments and observations made in an indoor multi-hop multi-radio 802.11b/g mesh networking testbed we established, *BilMesh*, we first analyze and model the nature of co-channel and adjacent channel interference. We conduct extensive experiments on this testbed to understand the effects of using multi-radio, multi-channel relay nodes in terms of network and application layer performance metrics. We also report our results on using overlapping in addition to orthogonal channels for the radios of the mesh routers. We then turn our attention to modeling and quantifying adjacent channel interference. Extending BilMesh with IEEE 802.15.4 nodes, we propose computational methods to quantify interference between channels of a wireless communication standard

and between channels of two different standards (such as Wi-Fi and ZigBee).

Majority of the studies in the literature on channel assignment consider only orthogonal channels for the radios of a multi-radio WMN. Having developed quantitative models of interference, next we propose two optimization models, which use overlapping channels, for the joint flow-radio and channel assignment problems in WMNs. Then we propose efficient centralized and distributed heuristic algorithms for coupling flows and assigning channels to the radios of a WMN. The proposed centralized and distributed schemes make use of overlapping channels to increase spectrum utilization. Using solid interference and capacity metrics, we evaluate the performances of the proposed schemes via extensive simulation experiments, and we observe that our schemes can achieve substantial improvement over single-channel and random flow-radio and channel assignment schemes.

Keywords: Multi-radio nodes, 802.11, 802.15.4, CSMA, TCP, UDP, Radio channels, Overlapping and orthogonal channels, Interference factor, Spectrum analyzer, Wireless mesh networks, Flow-radio assignment, Channel assignment, Distributed algorithms.

ÖZET

ÇOK-RADYOLU ÇOK-KANALLI KABLOSUZ ÖRGÜSEL AĞLARDA GİRİŞİMİN İNCELENMESİ VE AZALTILMASI

Alper Rifat Uluçınar
Bilgisayar Mühendisliği, Doktora
Tez Yöneticisi: Doç. Dr. İbrahim Körpeoğlu
Temmuz, 2013

Temel olarak, kablosuz bağlar ile birbirlerine bağlanmış örgüsel yönelticilerden oluşan omurga ağlar olan kablosuz örgüsel ağlar, tasarısız ağ oluşturabilme, öz-oluşum, öz-düzenleşim, öz-iyileşme gibi özelliklere sahip oldukları için kendilerine son mil geniş bant İnternet erişiminden olağanüstü durum ağlarına yahut eşler arası ağlara kadar çok geniş bir yelpazede uygulama alanı bulmaktadır. Kablosuz örgüsel ağların çoklu atlamalı doğası akışlar-arası girişimi arttırır ve akış-içi girişime sebebiyet verir. Bu etmenler de ağ kapasitesini ciddi ölçüde azaltır. Girişimi azaltıp ağ kapasitesini arttırmak için sıkça başvurulan bir yöntem örgüsel yönelticileri birden fazla iletişim kanalında çalışabilen birden fazla radyo ile donatmaktır. Böylelikle, bir örgüsel yönelticinin eş zamanlı olarak birden fazla kablosuz iletişim kanalını kullanması ve birden fazla kanal üzerinden koşut olarak paket alıp vermesi mümkün olmaktadır. Fakat birden fazla radyonun ve kanalın verimli olarak kullanılabilmesi için akış-radyo ve kanal atamayı da içeren dikkatli ve akıllı bir radyo kaynak planlaması gereklidir. Bu ise öncelikle, kablosuz örgüsel ağlar bağlamında kanal-içi girişimin ve komşu kanal girişiminin doğasını çözümlemeyi ve modellemeyi gerektirir.

Kanal-içi girişimi ve komşu kanal girişiminin etkilerini anlamak ve modellemek için, çok-radyolu 802.11b/g örgüsel yönelticilerden mütevellit, adını *BilMesh* koyduğumuz bina içi sınama ortamımız üzerinde deneyler ve gözlemler yaptık. Ayrıca, çok-radyolu örgüsel yönelticiler kullanmanın ve böylelikle çoklu atlamalı bir akışın ardışık atlamalarını farklı kanallardan geçirmenin ağ ve uygulama katmanı metrikleri üzerindeki etkilerini inceledik. Çok-radyolu örgüsel yönelticilerde sadece örtüşmeyen kanallar kullanmanın başarımını, örtüşmeyen kanalların yanında örtüşen kanallar kullanmanın başarımı ile mukayese ettik.

Daha sonra, komşu kanal girişimini modellemeye ve ölçmeye yöneldik. Bu amaçla BilMesh sınama ortamına IEEE 802.15.4 radyoları ekleyerek deneyler yaptık ve hem bir kablosuz iletişim standardının kanalları arasındaki girişimi hem de Wi-Fi ve ZigBee gibi farklı iki standardın kanalları arasındaki girişimi hesaplayabildiğimiz iki yöntem önerdik.

Literatürdeki kanal atama üzerine olan çalışmaların birçoğu, çok-radyolu kablosuz örgüsel ağlar için sadece örtüşmeyen kanalları kullanmaktadır. Girişim için nicel modeller geliştirdikten sonraki adım olarak, kablosuz örgüsel ağlarda birleşik akış-radyo ve kanal atama problemi için örtüşen kanalları da kullanan eniyileme modelleri önerdik. Daha sonra, yine birleşik akış-radyo ve kanal atama problemini çözmeye yönelik olarak, örtüşen kanalları da kullanabilen verimli merkezi ve dağıtık algoritmalar önerdik. Önerdiğimiz bu algoritmaların başarımını çeşitli gerçekçi girişim ve ağ kapasitesi metriklerini kullanarak, ayrıntılı benzetim modelleri ile gerçekleştirdiğimiz deneylerde ölçtük ve önerdiğimiz algoritmaların örgüsel kablosuz ağlarda tek kanal kullanarak veya rastgele yapılacak akış-radyo ve kanal atamaya göre büyük iyileşme sağladıklarını gözlemledik.

Anahtar sözcükler: Çok-radyolu düğümler, 802.11, 802.15.4, CSMA, TCP, UDP, Radyo kanalları, Örtüşen ve örtüşmeyen kanallar, Girişim çarpanı, Spektrum çözümleyici, Kablosuz örgüsel ağlar, Akış-radyo atama, Kanal atama, Dağıtık algoritmalar.

Acknowledgement

I would like to express my sincere gratitude to my supervisor Assoc. Prof. Dr. İbrahim Körpeoğlu for his endless support, guidance, invaluable contributions and patience in this difficult journey that led to this thesis. Without his continuous encouragement, this thesis would have never been completed.

I am also very grateful to Prof. Dr. Ezhan Karaşan for his contributions to my research and his wise suggestions. I am inspired a lot by him.

I would like to thank all committee members for accepting to read and review my thesis and for their constructive comments.

During my doctorate study, I have received financial support from the Computer Engineering Department of Bilkent University, the Scientific and Technological Research Council of Turkey (TÜBİTAK) and the European Union FP7 Programme (via Firesense Project). I would like to thank all these institutions for their support.

I owe special thanks to my colleagues at Bilkent University. I do appreciate their friendship.

This is a great opportunity for me to thank my wife, Özlem, who has shown great respect and understanding, and has supported me during my study. I would also like to thank my parents for their continuous support and encouragement.

To the precious memory of my aunt, Prof. Dr. Nur Günyaktı...

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Wireless mesh networking is an active area of research which is believed to be the next step in the evolution of the wireless architecture due to its relatively low cost, flexibility in the hardware and software options, ease of deployment, self-configuration and self-healing properties. Unlike ad hoc networks, infrastructure/backbone and hybrid wireless mesh networks (WMNs) employ a wireless mesh backbone composed of statically deployed mesh routers as an architectural component [1]. And similar to ad hoc networks, this backbone should be self-organizing and self-configuring for scalability, ease of deployment and ease of maintenance. In infrastructure/backbone WMNs, conventional clients (clients lacking the ability to forward packets on behalf of other nodes) access backhaul services and communicate with each other via the mesh backbone. The mesh backbone, therefore, provides mesh connectivity and routing services in a multi-hop manner for the conventional clients and other mesh clients.

Mesh networking paradigm provides better coverage and better scalability when compared with conventional wireless local area networks due to low deployment and low maintenance costs. Also since the capacity of a communication channel is logarithmically proportional to the signal-to-noise ratio (SNR) by Shannon's channel capacity formulation [2], and since increased deployment density implies increased SNR values in general, mesh networking paradigm can provide increased network capacities. Another advantage of the mesh networking

paradigm is that it can be applied by modifying layer-3 solely, which makes it possible to apply this paradigm on top of various wireless communication technologies such as Wi-Fi [3], WiMAX [4] or ZigBee [5], etc. No new hardware or software below layer-3 is required most of the time, which provides greater flexibility in hardware and software choices and decreasing costs.

One common approach when applying mesh networking onto wireless networking technologies that possess multiple overlapping or non-overlapping channels is to make use of multiple channels for adjacent hops. Some related studies following this approach are discussed in Section 2.2. This approach greatly reduces the hidden and exposed terminal issues, though does not completely annihilate them especially when overlapping channels are employed.

In order to be able to use multiple channels with the conventional Wi-Fi radios, one approach is to have the radios hop channels in the course of time [6, 7]. However, this approach requires temporal synchronization between the transmitter and receiver radios because the transmitter and the receiver must be operating on the same channel simultaneously to be able to communicate with each other. Hence, more complex transceivers are required. Another problem with this approach is the latency introduced to the system while switching from one channel to another.

Another approach to employ multiple channels on consecutive hops is to use nodes equipped with multiple radios [8]. Having multiple radios in each node allows assignment of different channels to adjacent links in the network. The channels can be assigned either statically or for long durations of time, and in this way, the radios do not need to perform channel hopping. Although currently available IEEE 802.11b/g hardware does not comprise multiple radios, it is possible to build a *logical* multi-radio node out of two or more single radio modules. This is the approach we pursue for our testbed and further details of this approach are discussed in Section 3.2.

Each radio of a multi-radio node can be configured to operate on a different channel so that packets arriving in the multi-radio node on one channel may

depart the node on a different channel. This scheme allows the packet transmissions on each hop of a (multi-hop) flow to be on different channels. If the channels used on the consecutive hops are overlapping channels, this causes *intra-flow* interference; meaning that the transmissions of a flow's packets on a specific hop interferes with the transmissions of the very same flow's packets on a consecutive hop.

To worsen the situation, there are usually other concurrent flows in the WMN. Transmissions of packets of different flows on different hops also interfere with each other (*inter-flow* interference).

Intra-flow and inter-flow interference degrade the network capacity severely. When a link is considered, interference from nearby links diminishes the SINR at the receiver. This results in an increase in BER and subsequently in PER, which further implies packet retransmissions at the various layers of the protocol stack.

Another factor diminishing the multi-hop network's capacity is the increased number of packet collisions. Transmissions on different hops, either belonging to the same flow or belonging to different flows, may collide with each other. In case of a multi-hop flow, at each hop, antecedent packet(s) of the flow will be in transmit queues waiting to be delivered to the next hop, while at the same time, the previous hop will be contending to deliver the following packets of the same flow. The stochastic nature of the commonly employed MAC protocols, such as the CSMA/CA, allows collisions in such a setting.

To mitigate interference in multi-radio multi-hop WMNs, majority of the existing studies prefer to use non-overlapping channels (see Section 2.2). However, the number of non-overlapping channels defined in a wireless communication standard can be limited as it is the case for the popular and widely deployed IEEE 802.11b/g. This has motivated the research community to investigate the possibility of using overlapping (in addition to non-overlapping) channels in multi-radio WMNs. Existing studies using this approach in the literature are surveyed in Section 2.3.

The first objective of the work introduced in this thesis is to analyze adjacent

and co-channel interference in the context of multi-radio multi-channel WMNs and to develop a quantitative model for the amount of inter-channel interference. The second objective is to propose centralized and distributed schemes for channel assignment that will intelligently mitigate interference and increase the capacity of WMNs.

To address the first objective, we first set up an indoor multi-hop multi-radio 802.11b/g mesh networking testbed and through extensive real-world experiments on this testbed, we analyze the nature of co-channel and adjacent channel interference in a multi-hop multi-channel setting. We investigate the effects of using overlapping channels in the consecutive hops of multi-hop flows on application and network layer metrics. We then extend our testbed with ZigBee radios and propose computational methods to quantify interference between channels of a wireless communication standard and between channels of two different standards. We report our measurements for the interference between IEEE 802.11b channels and between IEEE 802.11b and 802.15.4 channels.

To address the second objective, we first develop optimization models for jointly handling the flow-radio assignment and channel assignment problems. These models use overlapping channels for assignment and incorporate the effects of an idealized MAC protocol in their formulations. Then we propose centralized and distributed heuristics that efficiently address the same problems as the optimization models. The proposed centralized and distributed schemes make use of overlapping channels to increase spectrum utilization.

In our optimization models and centralized and distributed schemes, we consider the channel assignment problem in relation with the flow-radio assignment problem. We call this joint handling of the flow-radio assignment and channel assignment problems as the joint flow-radio and channel assignment (JFRCA) problem.

1.1 Contributions of the Thesis

This thesis has both practical and theoretical contributions. In the beginning of each chapter, we detail the major contributions made in that chapter. Below, we give a brief summary of these contributions by classifying them into two categories.

1. *Practical Aspects:* The major practical contributions of this thesis are presented in Chapters 3 and 4. In Chapter 3, we introduce our multi-radio WMN testbed, BilMesh, and in Chapter 4, we present experimental, measurement-based methods for quantifying interference. In these chapters, we draw important conclusions on the multi-hop nature of WMNs and on the operation of the CSMA/CA MAC under adjacent channel interference. We report our interference factor measurements between IEEE 802.15.4 and 802.11b channels. The work presented in these two chapters lays the foundation of the theoretical and algorithmic work presented thereafter.

2. *Theoretical Aspects:* We present mathematical models and centralized and distributed algorithms for flow-radio coupling and channel assignment in Chapters 5-7. These works constitute the theoretical aspects of this thesis and are based on the practical results of the previous chapters.

1.2 Thesis Outline

In Chapter 2, we introduce the key concepts that this thesis is based on and give preliminary background information on the subjects studied in this thesis. We also give critical reviews of the literature on WMN testbeds, interference factors and channel assignment algorithms.

In Chapter 3, we introduce our indoor 802.11b/g mesh networking testbed (BilMesh) established in Bilkent University. We describe the testbed's architecture and configuration in detail and present our novel multi-radio node architecture. We perform extensive sets of experiments on BilMesh to investigate the

multi-hop nature of WMNs. We report our measurements on various application and network layer metrics. We also perform experiments to investigate how the wireless channel separation between the subsequent hops of a (multi-hop) flow affects the achievable goodput and other network layer metrics, such as delay and jitter. Another critical issue we investigate in Chapter 3 is the performance of the CSMA/CA MAC in the existence of adjacent channel interference, especially when the interference comes from an overlapping channel.

In Chapter 4, we focus on the concept of interference factor (I-factor) and propose two new methods for measuring and obtaining the interference factors between the channels of a wireless technology. The flexibility in our methods allows them to be used for also measuring and obtaining the interference factors between channels belonging to different wireless technologies. We report our interference factor measurements among IEEE 802.11b DSSS channels and also between IEEE 802.15.4 and 802.11b channels.

Having investigated the multi-hop multi-channel nature of the WMNs and having quantified the interference between the channels of the widely deployed 802.11b technology, in Chapter 5, we turn our attention to the joint flow-radio and channel assignment (JFRCA) problem in the context of multi-radio WMNs, and we propose two flow-aware optimization models that also incorporate the effects of MAC protocols. Using these mathematical models, we further analyze interference and the relation between distance and link capacities under adjacent channel interference on exemplary network topologies.

Then, in Chapter 6, we propose centralized algorithms that address the joint flow-radio and channel assignment problem. The NP-hardness of the channel assignment problem in the context of multi-radio WMNs motivates us in developing these centralized heuristic schemes. The proposed centralized schemes make use of the overlapping channels in addition to the available orthogonal channels. In Chapter 6, we also propose novel metrics for assessing the amount of average interference and the residual capacities of the receiver radios. We evaluate the performance of the proposed schemes using random topologies and discuss our results.

In Chapter 7, we develop the notion of interference subgraphs and address the joint flow-radio and channel assignment problem in the framework of a flow-aware distributed protocol. We propose a distributed scheme that can assign flows and channels to radios in a distributed decentralized manner. Our distributed scheme consists of many sub-algorithms and we describe these distributed algorithms in every detail. We also implement a discrete-event simulation model of our proposed scheme. We first validate our distributed scheme on some small topologies (for which it is easy to compute optimal solutions), and then we perform extensive simulation experiments on random grid topologies of greater size (in terms of multi-radio node counts, number of radios per node and number of flows) to assess its performance.

Finally in Chapter 8, we conclude the thesis and point to some possible research directions related with the work presented in this thesis.

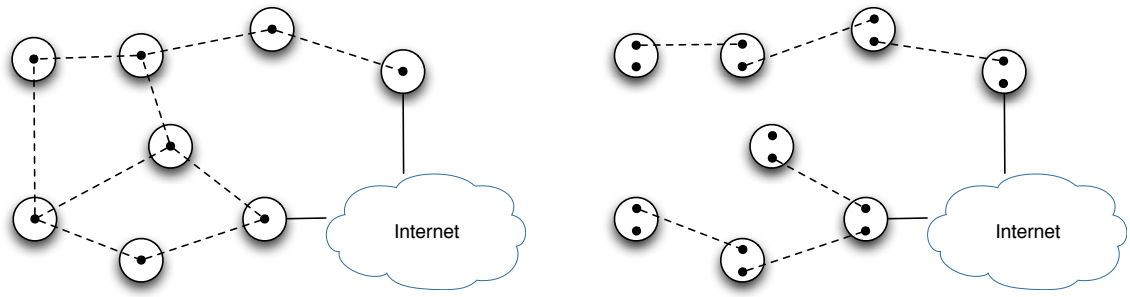
Chapter 2

Background and Related Work

In this chapter, we introduce the reader with a minimal background of the concepts utilized in this thesis and give a critical review of the related literature. We begin our discussion with a brief survey of the various academic and community WMN deployments and testbeds in Section 2.1. Then we discuss the key concepts of *overlapping* and *non-overlapping* channels together with the concept of interference factor in Section 2.2. Finally, we arrive at the discussion of the flow-radio assignment and channel assignment problems in the context of multi-radio multi-channel WMNs in Section 2.3.

2.1 Wireless Mesh Network Deployments and Testbeds

In this section, we first briefly review the architectures of single radio and multi-radio WMNs. Then we provide a brief summary of some of the available mesh networking platforms and the related work done in multi-radio multi-channel WMNs. Most software choices in the platforms mentioned here are available in source code from their developers and operate on a variety of hardware. Most common choices run on Linux and Microsoft Windows operating systems.



(a) Single-radio mesh routers.

(b) Two-radio mesh routers.

Figure 2.1: Conceptual diagrams of single-radio and multi-radio infrastructure WMNs.

Figure 2.1 shows the conceptual diagrams of single-radio and two-radio WMNs. In both diagrams, solid lines represent wired communication links (such as the Ethernet) and dashed lines represent wireless links (such as IEEE 802.11n links). Filled small circles are the radio interfaces. The enclosing circle around a radio interface (or around a group of two interfaces in Figure 2.1(b)) represents a *mesh router*. To simplify the discussion without loss of generality, in Figure 2.1 we assume that the wireless links are symmetric; node j can receive packets from node i if and only if i can also receive packets from j.

A good survey on WMNs can be found in [1]. The multi-hop network formed by the mesh routers provides an infrastructure for the client nodes. Some of the mesh routers also act as *access points*, so that client nodes can attach to the WMN via these mesh access points (MAPs). The two mesh routers in Figure 2.1 with the wired links act as *gateways* to Internet. In a typical infrastructure WMN, traffic is directed towards these gateway nodes. However, traffic patterns in a WMN depend on the applications running in the network. In our study of WMNs, we make no assumptions on the applications running in the network.

In Figure 2.1(a), each mesh router is equipped with a single radio. Each radio is operating on the same wireless channel. Hence, the graph in Figure 2.1(a) represents the connectivity graph (i.e., if mesh routers i and j are in the transmission range of each other, then a wireless link between them exists). However, as seen in Figure 2.1(b), not every possible link (considering internodal distances) has

been established. This is due to the fact that not all radios are operating on the same channel. Despite this, as in Figure 2.1(a), each mesh router can still relay its packets towards one of the gateway nodes in a multi-hop manner.

We now review the major single-radio and multi-radio WMN deployments and testbeds of various scales in the literature, together with the issues raised by the researchers about them. We also give an overview of the commonly available software and hardware alternatives to establishing a WMN. Some of these deployments have solely research motivations whereas some others solve real-world problems, such as sharing broadband access across a campus. We discuss our multi-radio WMN testbed in detail in Chapter 3.

MIT CSAIL *Roofnet* [9] is an experimental mesh network developed at the MIT CSAIL and deployed over a 4 km^2 region providing broadband Internet access to its nodes. The average internode throughput is reported to be 627 Kbps for 37 nodes. Roofnet runs in a pseudo-IBSS mode which omits 802.11 beacons and BSSID mechanism. The main functionality provided by Roofnet is broadband Internet access and not peer-to-peer connectivity. Roofnet software is distributed in multiple choices: as a firmware for Netgear WGT634U access points, as a live CD distribution which contains a 45 MB Linux image compiled for the i386 architecture and as an OpenWRT 2.0 package. Roofnet uses Srcr [9] as its routing protocol, and SampleRate [10] as its rate selection algorithm.

Microsoft Research's Mesh Connectivity Layer (MCL) [11] is part of *Microsoft's Mesh Networking Academic Resource Toolkit* and is also available as a stand-alone download both in binary and source code forms. The toolkit includes MCL source code for Windows XP and Windows CE together with performance measurement tools, configuration tools and related documentation and publications. MCL is a loadable Windows driver which implements a virtual network adapter. MCL sits between the data link layer and the network layer and implements ad hoc routing with link quality measurements. The routing algorithm is Multi-Radio Link Quality Source Routing (MR-LQSR) [12], which is a modified version of DSR. MCL can utilize multiple wireless adapters operating at different channels and hence can be used to drive multi-radio architectures. One

limitation of the software is that, in the case of multi-radio systems, the radios should be driven using different device drivers. MCL is a good alternative for those wishing to operate their wireless mesh network on the Windows platform.

JHU DSN *SMesh* [13] is an 802.11 mesh network deployed at the Distributed System and Networks Lab at Johns Hopkins University. It provides peer-to-peer connectivity, Internet connectivity and fast handoff to mobile VoIP clients. SMesh operates in standard IBSS mode. Mobile clients send and receive data through the mesh infrastructure provided by SMesh and do not rely on each other for forwarding packets. The multi-hop communication infrastructure used by SMesh is provided by Spines [14, 15], which is developed by the same group. Spines provides a generic multi-hop messaging infrastructure that allows unicast, multicast and anycast communication with an API similar to the Unix sockets. SMesh binaries are provided upon e-mail request [16]. It is reported on the SMesh Internet site that it has been tested on x86 architectures and on Linksys WRT54G routers.

In [17], Robinson et al. investigate the limitations of the multi-radio testbed platforms and quantify the impacts of specific platform choices only on the application layer throughput. Their wireless mesh testbed is a 2-hop network consisting of a workstation equipped with multiple PCI 802.11b cards. They identify three main causes of performance degradation: Board crosstalk, RF power leakage and inadequate separation between Wi-Fi antennas. They also try to mitigate PCI board crosstalk by shielding the Wi-Fi cards with aluminium foil. Similar observations about board crosstalk have been made in [8] and in [12]. In [18], Zhang et al. set up a cabled wireless testbed with two PCs. Each of the PCs are equipped with up to 4 802.11a NICs and all NICs are interconnected by couplers and attenuators through a splitter in order to eliminate all wireless medium related factors. Their aim is to study CPU utilization and the effects of board crosstalk between PCI NICs. They report that, for an 802.11a network in a saturated network condition, computing resources is the key limiting factor on the performance rather than the crosstalk between the PCI Wi-Fi cards.

In existing multi-radio mesh networking testbeds, multi-radio nodes are built

using multiple PCI or mini-PCI Wi-Fi NICs installed in a single computer system. As the previous studies mentioned above have shown, due to board crosstalk on a single multi-radio system built using commodity hardware, multi-hop network performance is severely degraded. In order to be able to completely eliminate the adverse effects of board crosstalk, we take a different and novel approach in the design of our multi-radio nodes, which is discussed in detail in Chapter 3. Two physically separate single-radio APs connected with a high speed wired link constitute our multi-radio node. This approach also scales well with the increasing number of Wi-Fi radios of a multi-radio node because each additional Wi-Fi radio of a node comes with its own CPU and main memory. With this multi-radio node architecture, we also have the flexibility to spatially separate the Wi-Fi antennas as needed. Unlike previous testbeds, we can also more effectively address the issues caused by RF power leakage by separating the antennas of the multi-radio node spatially and RF shielding them. In some of the experiments discussed later in Chapter 3, we have separated the two antennas of the two-radio nodes and shielded RF radiation, from each other using panels covered with aluminium foils. Another key difference between our multi-radio WMN testbed and the previous testbeds mentioned above is that we are using OLSR as the routing protocol in a multi-radio setting.

2.2 Wireless Communication Channels and Interference Factors

Wireless communication standards, such as the IEEE 802.11 family of standards, divide the allocated RF spectrum into *channels*. Some of these predefined channels share, in part, the same frequency band (i.e., they *overlap*) and some channels do not have any frequency band in common (they are *orthogonal*). Each channel has a predefined *center frequency* and a frequency width (bandwidth), both specified by the standard. The bandwidth required for a channel depends on, among many other factors, the modulation technique adopted. For example, if a spread spectrum method is adopted, then the required bandwidth will be significantly

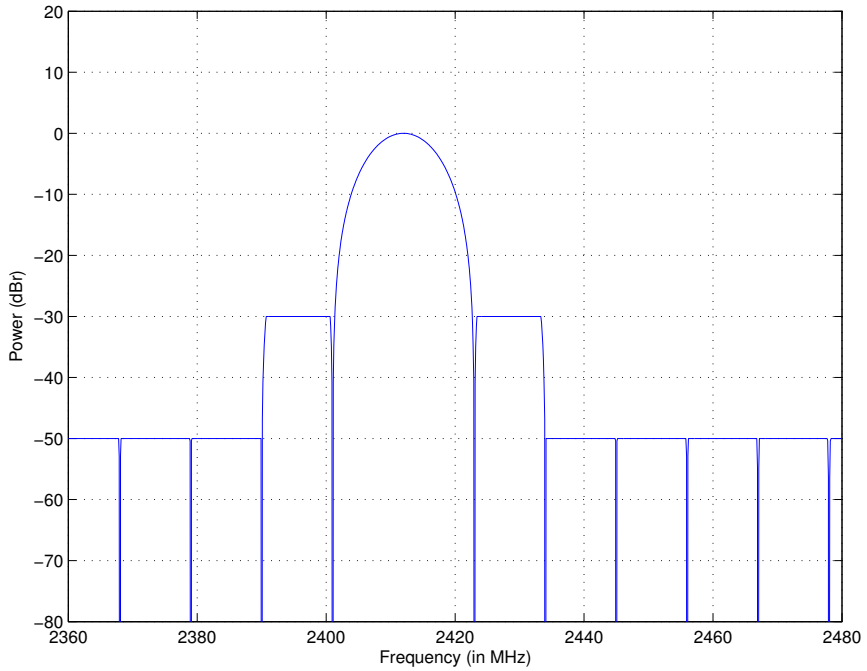


Figure 2.2: Filtered DSSS power spectral distribution. Center frequency is 2412 MHz.

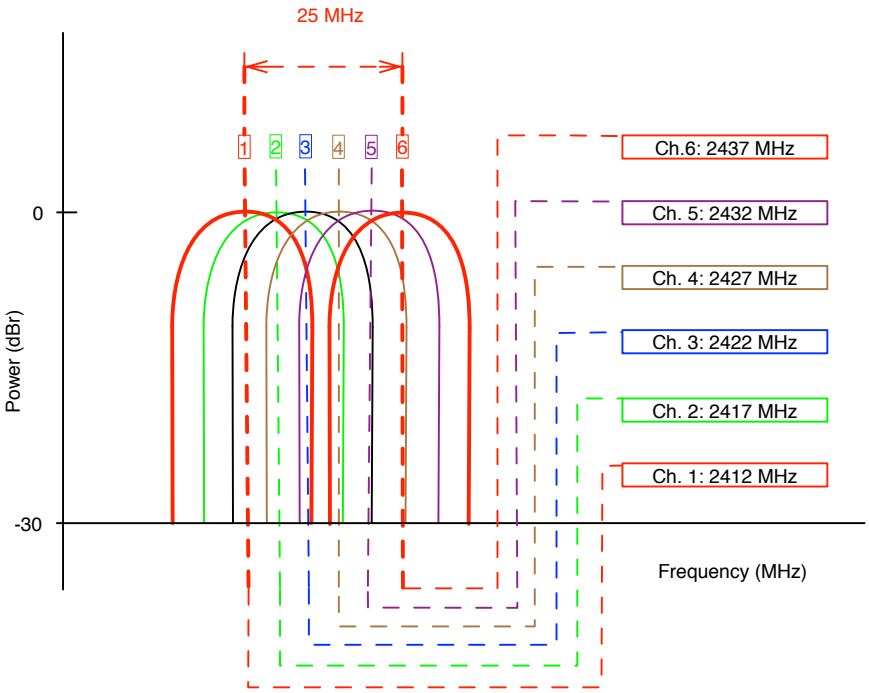


Figure 2.3: The first six channels of 802.11b/g.

larger than the information rate [19].

The initial revision of the IEEE 802.11 PHY specification in 1997 defines a PHY layer based on the direct sequence spread spectrum (DSSS) modulation technique [20]. The 802.11b DSSS based PHY defines data rates of 5.5 and 11 Mbps. DSSS is a spread-spectrum technique, and so it possesses the advantages of spread-spectrum based modulation: frequency diversity and ease of distributed coordination of multiple access [19, 21].

The 802.11 DSSS transmitter uses a *transmit spectrum mask* (TSM) [22] to suppress transmission power that leaks outside its 22 MHz band (see Section 4.4.1). Figure 2.2 shows the power spectral distribution of the filtered DSSS signal. In the IEEE 802.11b/g PHY specifications, there are 11 channels (in the FCC domain), where each channel is 22 MHz wide and the central frequencies of consecutive channels are separated by 5 MHz. When the center frequencies of two channels are separated by more than 22 MHz, these channels are considered to be non-overlapping (i.e., orthogonal) channels [23]. In 802.11b/g, for two channels to be considered as non-overlapping channels, they should be at least 5 channels away from each other, because 5 channels of separation implies that the channel center frequencies are separated by 25 MHz, which is greater than 22 MHz. Otherwise, if two channels are separated by less than 5 channels, they are overlapping. Hence, channels 1 and 6, for example, are non-overlapping whereas channels 1 and 5 are overlapping (see Figure 2.3). There are at most 3 non-overlapping channels (channels 1, 6, and 11) in IEEE 802.11b/g that can be used simultaneously. In this thesis, we use the terms non-overlapping channels and orthogonal channels interchangeably.

The concept of *interference factor* [23–25] has been developed to quantify, between 0.0 and 1.0, the amount of overlap and interference between adjacent channels. Assuming x and y are two channels defined by a wireless communication standard, if the interference factor between x and y, $I(x, y)$, is 0.0, then there is no overlap (in the frequency domain) between these two channels. If $I(x, y) = 1.0$, then these two channels occupy the same frequency band ($x = y$). As an example in Figure 2.3, $I(1, 1) = 1 > I(1, 2) > I(1, 3) > I(1, 4) > I(1, 5) > I(1, 6) = 0.0$.

There are two main classes of interference factor (*I-factor*) models in the literature. The first class comprises analytical models [23, 25, 26], which are generally applied to relatively simple modulation techniques, such as the DSSS, because of the complexities of the models. The second class comprises a set of experimental measurement-based methods [23, 24], which are more flexible than the analytical methods because they are not built upon the specifics of a physical-layer technique; they involve measurements in any of the various layers of the Open Systems Interconnection (OSI) stack [27].

One of the early works on I-factor belongs to Mishra et al. [23]. In this study, the authors propose the I-factor concept to model the amount of transmit power radiated by a transmitter on channel j and received by a receiver on channel i. They propose both an analytical model which allows theoretical values to be calculated for the I-factor between two given 802.11b DSSS channels and an empirical model based on throughput measurements. In [26], Villegas et al. give a good analytical account of adjacent channel interference in the contexts of DSSS and OFDM systems.

Mishra et al. [24] discuss how partially overlapping channels can be leveraged to improve spatial channel reuse in Wireless LANs. Through experiments, they quantify, as a function of the physical data rate, the interference range of an Access Point (AP) - Station (STA) pair with respect to another AP-STA pair operating on an overlapping channel. In the context of single-radio mesh networks, the authors also investigate the possibility of receiving data from a transmitter operating on an overlapping channel with respect to the receiver's channel.

The most direct and more commonly adopted experimental method of obtaining an I-factor model is to perform Signal-to-Noise Ratio (SNR) [2] measurements. In these models, a receiver is kept fixed at a channel and its transmitter is operated on non-overlapping and overlapping channels. For each channel of the transmitter, SNR is measured on the receiver and normalized to a scale of $[0, 1]$ as in [24]. This method mandates that the interferer (transmitter) and the receiver must be using the same wireless communication standard, so that SNR readings (where the signal belongs to the interferer) are available at the receiver.

If the interferer uses a different wireless communication standard than the receiver (such as the interferer being a Bluetooth radio and the receiver being an 802.11b/g radio), then there will be no links between these two radios and no SNR measurements will be available at the receiver radio.

Feng and Yang [28, 29] use numerical methods to analyze network capacity improvements that can be gained by using partially overlapping channels. While defining the carrier sensing range between two nodes operating on channels i and j, they perform a set of testbed experiments that involve two pairs of nodes. One pair communicates with each other on channel i and the other pair communicates on channel j. The authors define the carrier sensing range as "the maximum distance that these two can affect each other's communications" [29]. Then they give statistical and numerical models of capacity improvements when overlapping channels are used compared to using only orthogonal channels in one-hop and multi-hop wireless networks. In [29], the authors also discuss the cases where no improvement can be gained by using partially overlapping channels.

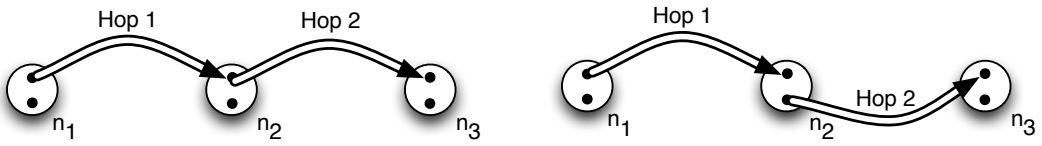
Zhou et al. [30] envision that in the very near future, the world will be full of low-power wireless sensors sharing the same spectrum. As an illustrative example, they measure the 2.4 GHz spectrum with their HP 8593E spectrum analyzer in the coexistence of a microwave oven, a cordless 2.4 GHz presenter, and a MICAz sensor network. They also report the reception ratios of the MICAz motes when the microwave oven is on and when it is off. However, they do not model interference using these measurements. The authors propose the dimensions along which new wireless sensor network protocols should be designed to cope with the crowded spectrum issue.

Fuxjäger et al. [31] pose the fundamental question of whether there really is no interference between the non-overlapping channels of IEEE 802.11. To investigate this, the authors use a testbed consisting of four laptops, each equipped with an Intel PRO 2200BG mini-pci card and running Linux. They place the laptops on a linear line-of-sight topology, each raised 1.5 m above the ground. Using this testbed, the authors measure the MAC and transport layer throughputs and MAC frame loss ratios. They also measure the goodput of a TCP flow. The

authors conclude that due to the near-far effect [32], cross-channel interference exists between non-overlapping channels of IEEE 802.11 when the receiver and the interferer radios are placed only tens of centimeters away from each other. They also conclude that off-the-shelf IEEE 802.11 chipsets may not be ready to be placed in the same box for use in multi-radio wireless mesh networks.

Petrova et al. [33] investigate the performance of IEEE 802.15.4 networks under the interference caused by IEEE 802.11g and pre-standard IEEE 802.11n networks through measurements. They use a testbed consisting of an 802.11g/n access point, a laptop used as the 802.11g/n traffic sink and equipped with an 802.11g/n adapter, a PC used as the 802.11g/n traffic generator, and two TelosB motes. They also monitor the 2.4 GHz spectrum with an Agilent E4440A spectrum analyzer. Using this testbed, the authors measure the packet delivery ratios of the 802.15.4 network. They use the spectrum analyzer to report the average power spectral densities of the 802.11n signals for different alignments of the 802.11n nodes. However, they do not model interference using these measurements.

In Chapter 4, we propose two physical-layer-measurement-based methods for calculating I-factor values. Unlike previous work, our methods are generic enough to model the interference between channels of any two wireless communication technologies, i.e., they can be used to calculate the I-factor values between the channels of a wireless technology and between the channels of two different technologies (such as the IEEE 802.11b and IEEE 802.15.4). Also, these methods are capable of quantifying the interference from non-communication devices. We perform measurements on our testbed, and in Chapter 4, we report the I-factor values between 802.11b channels and between 802.11b and 802.15.4 channels.



(a) Alternative 1.

(b) Alternative 2.

Figure 2.4: Two alternative flow-radio couplings for a two-hop flow from n_1 to n_3 via n_2.

2.3 Flow-Radio and Channel Assignment in Wireless Mesh Networks

Through theoretical and practical methods, researchers have quickly realized that WMNs with single-radio nodes have severely limited capacities due to the interference intensified by the multi-hop nature of these networks [9,34,35]. Multi-hop flows cause intra- and inter-flow interference in a WMN, and there is also interference from foreign wireless networks operating in close proximity of a WMN.

A widely accepted approach to mitigate intra- and inter-flow interference is to equip the mesh nodes with multiple radios that support multiple frequencies (channels) so the radios can be tuned to different channels. Consider the multi-hop flow in Figure 2.4 from n_1 to n_3 in a multi-radio WMN. Assuming we are given which multi-radio nodes it will visit en route, we ask the following question: On each node the flow visits, which radio of the node will the flow use, i.e., be coupled with? In other words, given the route, what will be the *flow-radio assignments*?

The flow depicted in Figure 2.4 has a total of 2^3 possible arrangements for flow-radio coupling (flow to radio assignment). In Figures 2.4(a) and 2.4(b), two of these are shown. For two radios to communicate reliably with each other, they must be tuned to the same wireless channel. In Figure 2.4(a), both hops of the flow must be on the same channel; whereas in Figure 2.4(b), the first hop and the second hop of the flow can be on different channels. Assuming the routes are given a priori, flow-radio assignment determines which radio pairs will be used to carry flows, hence which links should be established between multi-radio nodes.

Given the half-duplex operation of the radios, in Figure 2.4(a), n_2 will not be able to send a packet to n_3 on the second hop while it is busy receiving a packet from n_1 on the first hop. On the other hand, in Figure 2.4(b), n_2 can send and receive packets on both hops in parallel. However, if the two hops in Figure 2.4(b) are on the same channel, transmissions on the second hop will severely interfere with the receptions on the first hop.

So intelligent channel planning is necessary while determining channels for the hops (and correspondingly for the endpoints of the hops, i.e., the transmitter and receiver radios). Now, we pose our second question: Which channels should be assigned to the radios utilized by flow-radio assignment? Or in other words, which channels should be assigned to the radios on which at least one flow is coupled? The flow-radio assignment and the channel assignment problems in the context of multi-channel multi-radio (MC-MR [36]) WMNs are tightly coupled. In Chapters 5-7, we deal with these two problems in a joint manner, and we call the joint problem as the joint flow-radio and channel assignment (JFRCA) problem. In this thesis, we use the terms flow-radio coupling and flow-radio assignment interchangeably.

Vast majority of the existing literature on channel assignment in multi-radio WMNs uses only non-overlapping channels and very few studies consider flow-radio assignment. Due to the limited number of orthogonal channels in the IEEE 802.11b/g standards, researchers have also investigated the possibility of using overlapping channels. The multiple subset sum problem can be reduced into the channel assignment problem as shown in [37], which proves that the channel assignment problem in the context of multi-radio WMNs is NP-hard.

Existing literature on the channel assignment problem can be broadly classified into three categories: centralized algorithms, mathematical models and distributed algorithms. We first outline the mathematical models in the literature addressing this problem and then discuss the centralized and distributed algorithms.

In [25], the authors extend the linear programming (LP)-based formulation of [38], which performs joint channel assignment and routing in multi-radio

WMNs, to use partially overlapped channels as well as non-overlapping (orthogonal) channels. They demonstrate via simulations that the use of partially overlapping channels in the contexts of Wireless LANs and multi-hop Wireless Mesh Networks can improve end-to-end application throughput.

In [39], Rad et al. propose an optimization model (*JOCAC*) that is solved by exhaustive search for joint channel assignment and congestion control of TCP traffic in an infrastructure multi-radio WMN. The solution to the model is searched exhaustively either in a centralized manner on a gateway node to yield an optimal solution, or in a distributed manner on each multi-radio node to yield a partially optimal solution. JOCAC assumes a tree routing topology like [40] and does not address the flow-radio assignment problem in a setting where the traffic does not concentrate on gateway nodes.

Both [41] and [42] propose mixed integer linear programs (MILP) for the joint channel assignment and flow-radio assignment problem, and use partially overlapping and orthogonal channels. In [41], the proposed formulation incorporates network traffic information and is load aware, with the objective to maximize aggregate end-to-end throughput while minimizing queuing delays.

With its problem domain specification the joint flow-radio and channel assignment problem, and with its load aware formulation, the work in [41] is the closest to ours. However, Bukkapatanam et al. propose a load aware MILP formulation in [41], whereas in Chapters 6 and 7, we propose a set of centralized and distributed tunable heuristic algorithms for the same domain. Hence, our schemes can scale better and work for larger networks efficiently.

In [43], Ramachandran et al. propose a centralized algorithm (called *BFS-CA*) for channel assignment in multi-radio WMNs to minimize interference from co-located wireless networks. They define an interfering radio with respect to a multi-radio node of the WMN as a simultaneously operating radio visible to the WMN node but external to the WMN, and estimate interference on a specific channel with the number of interfering radios on that channel.

In [44], Skalli et al. propose an interference-minimizing centralized channel

assignment scheme (called *MesTiC*) that considers traffic patterns of the mesh network and connectivity issues. Like [43], MesTiC relies on using a default channel for topological connectivity and network management purposes. MesTiC assumes that WMN traffic is directed towards a gateway node that provides access to the wired network.

Another centralized algorithm specific to the infrastructure multi-radio WMNs, where the outgoing traffic is directed to a gateway node, is *POCAM* [45] (Partially Overlapped Channel Assignment for MRMC-WMN). POCAM is a backtracking search algorithm for channel assignment and does not address the flow-radio coupling problem. POCAM assumes a tree routing topology rooted at the gateway node.

In [46], Hoque et al. propose a new interference model derived in a broad sense from the I-factor [25] model of Mishra et al., and propose the concept of the *I-Matrix*. I-Matrix is a table maintained separately for each multi-radio node of the WMN. Each row of the I-Matrix holds the interference effects (costs) from all other channels for a specific channel. Using the I-Matrix tables, a centralized load-aware channel assignment algorithm which iteratively assigns channels to the links is proposed. The proposed algorithm makes use of the partially overlapped channels. As a channel is assigned to a link, the I-Matrices of all of the multi-radio nodes are updated. The flow-radio coupling problem is not addressed.

In [40], Raniwala et al. propose a multi-channel WMN architecture (called *Hyacinth*) based on nodes equipped with multiple 802.11 radios and the associated distributed channel assignment and routing algorithms. Hyacinth's 802.11 interfaces operate on non-overlapping channels and the distributed channel assignment algorithm assumes that the connectivity graph of the multi-radio nodes is a tree, which implies similar assumptions with [44]. The flow-radio coupling problem is again not addressed. The centralized and distributed heuristic algorithms proposed in Chapters 6 and 7 make no assumptions on the traffic patterns of the WMN and address the flow-radio coupling problem jointly with the channel assignment problem.

In [47], Subramanian et al. develop semi-definite programming (SDP) and integer linear programming (ILP) models to obtain bounds on the optimal solution of the channel assignment problem using orthogonal channels, and they generalize their ILP model for overlapping channels. They propose a Tabu search-based centralized algorithm and another centralized algorithm based on a greedy heuristic for the Max K-cut problem. Without considering the flow-radio assignment problem or the network traffic patterns, they derive a greedy distributed algorithm from the centralized Max K-cut based one.

In [48–50], distributed schemes for jointly addressing channel assignment and routing in multi-radio wireless networks are proposed. The distributed scheme proposed in [51] considers only the channel assignment problem. Common to [48–51] is that they only use orthogonal channels for channel assignment and do not consider the flow-radio assignment problem.

In [52], a cluster-based topology control and channel assignment algorithm ($CoMTaC$), which is based on the usage of default radio interfaces operating on default channels, is proposed. Each cluster selects its default channel by passively monitoring the traffic load on each channel as in [43]. A multi-radio node bordering multiple clusters has its second interface tuned to the default channel of the highest priority neighbor cluster. For selecting the channels of the non-default radio interfaces, each node estimates the interference on each channel using the average link layer queue length as an interference metric. CoMTaC does not address the flow-radio assignment problem.

Ko et al. in [53] propose a distributed channel assignment algorithm and the accompanying distributed protocol for multi-radio 802.11 mesh networks. They employ a greedy heuristic for channel selection that uses only local information and do not consider flow-radio assignment or routing. They do not use network traffic information and perform channel assignment using only physical topology information. Similar to the I-factor concept, they model interference between wireless channels using a linear cost function $f(a, b)$ (a and b being the wireless channels) and use overlapping channels.

In Chapter 5, we propose optimization models that address the joint flow-radio and channel assignment problem. We take the effects of an idealized MAC protocol into account and we use overlapping channels in addition to orthogonal ones for channel assignment in these models. We propose centralized and distributed heuristics that use overlapping channels, respectively in Chapters 6 and 7 for the joint flow-radio and channel assignment problem. We also introduce novel metrics for assessing the average interference and the residual capacities of the receiver radios in Chapter 6. Despite its prominent impact on the efficiency achievable by channel assignment, previous studies in the literature have overlooked the flow-radio assignment problem. Our work in these chapters is amongst the first to jointly address these two problems.

Chapter 3

BilMesh: A Multi-Radio Multi-Hop Wireless Mesh Networking Testbed

We have established an indoor multi-hop multi-radio 802.11b/g mesh networking testbed at Bilkent University, called *BilMesh*, for observing and studying the nature of multi-hop multi-radio communications as well as the nature of multi-hop single-radio communications in wireless mesh networks. In this chapter, we describe BilMesh in detail. We provide details about how a multi-radio mesh network that supports ad hoc routing can be built and configured using commodity hardware and software, together with the details of our node architecture, software configuration and network topology. We also report about our performance experiments conducted on multi-hop topologies with single-radio and multi-radio relay nodes in this testbed. We investigate and report the effects of using multi-radio, multi-channel relay nodes in the mesh networking infrastructure in terms of network and application layer performance metrics. We also study the effects of physical channel separation on achievable end-to-end goodput perceived by the applications in the multi-radio case by varying the channel separation between the radio interfaces of a multi-radio relay node.

In Section 3.1, we discuss our main motivations in establishing a multi-hop multi-radio wireless mesh networking testbed and list our major contributions. In Section 3.2, we describe BilMesh and its architecture in detail. Section 3.3 covers the descriptions and performance measurement results for multi-hop topologies with single-radio and multi-radio relay nodes (mesh routers). In Section 3.4, we conclude the chapter.

3.1 Introduction

Various non-academic communities have built urban wireless mesh networking infrastructures using low cost commodity hardware and open software. Also many academic groups have reported establishing wireless mesh networking testbeds to research various issues related with the paradigm. Since the mesh networking paradigm is generally applied onto existing MAC and physical layers and is used in conjunction with the widely adopted transport layer protocols, such as TCP, that are not capable of appropriately dealing with packet losses occurring in multi-hop wireless links, researchers are faced with many challenges originating from the MAC and transport layers while designing wireless mesh networks. The multi-hop nature of the wireless mesh backbone and the shared/broadcast nature of the wireless medium also give rise to the well-known hidden and exposed terminal issues. Another important issue arising from the broadcast nature of the wireless medium is that packets of the same multi-hop flow interfere with each other while traversing subsequent links. We established BilMesh testbed to study these issues. As we clearly show through experiments conducted in our testbed, this intra-flow interference greatly destabilizes multi-hop flows and reduces achievable goodput.

Most existing studies in the literature that deal with the channel assignment problem in the context of multi-radio multi-channel WMNs consider only non-overlapping channels. However, as surveyed in Section 2.3, works of Mishra et al. [25] and others have demonstrated via simulations that using overlapping channels in addition to the orthogonal (non-overlapping) channels can actually

improve end-to-end application throughput. With its novel, flexible multi-radio node architecture that provides elasticity in antenna placement and that can effectively deal with the Wi-Fi NIC related crosstalk issues, BilMesh is an attempt to further investigate these problems and explore the limitations arising in realistic settings.

We have set up networks of up to seven hops using single radio nodes and up to four hops using multi-radio nodes, which reveal previously unobserved facts about the relative performance of TCP and UDP in a multi-hop multi-radio setting. Using our testbed, we first investigate and report on the performance improvements achievable by using orthogonal (non-overlapping) channels for consecutive wireless hops. Then, we quantify by a set of extensive experiments, the goodput gains of using partially overlapping channels instead of using only orthogonal channels which is the method commonly followed in the literature. We look at the effects of different channel combinations and permutations on the performance that a multi-hop network flow experiences in terms of throughput and packet loss rate. In our study, we also investigate how carrier-sense based multiple access mechanism performs if the carrier sensing radio is operating on a different channel than the actively transmitting radio.

Our main contributions in this chapter are:

- Through BilMesh, we describe in detail how a single and multi-radio wireless mesh network can be built, established and configured with dynamic routing using off-the-shelf 802.11 wireless routers. We report our own experiences with BilMesh, which can be useful for other researchers who want to establish mesh networks.

- We propose a novel, cost-effective multi-radio node architecture for wireless mesh networking testbeds that is flexible in terms of number of radios, antenna placement and RF shielding. Our multi-radio node architecture also does not have the Wi-Fi NIC related crosstalk issues and scales well with the increasing number of Wi-Fi radios since the amount of available computing resources increases with the number of radios.

- Unlike previous multi-radio mesh networking testbeds, BilMesh uses OLSR [54] as its routing protocol. The OLSR protocol implementation we use in BilMesh is olsrd [55]. We discuss the details of the configuration of olsrd in a multi-radio setting.

- We investigate the effects of physical channel separation on the performance of a wireless mesh network with single and multiple radios. Effects on the network layer parameters such as average delay and delay jitter as well as on the transport layer performance (throughput and goodput) are studied.

- We observe that, although UDP is believed to perform better than TCP in terms of achievable goodput, and is thus generally chosen as the transport protocol for multimedia applications which require high bandwidth, this is not always the case in multi-hop wireless networks. As the number of hops a traffic flow traverses increases, TCP begins to achieve higher goodput than UDP. Hence, we propose that if no flow control is implemented at the application level, the transport layer protocol for multimedia applications should be chosen as a function of the number of hops multimedia packets have to traverse.

- We observe and report, by the results of our detailed experiments, that in a multi-hop UDP flow, round trip times for packets increase almost linearly with increasing hop count, whereas jitter increases almost exponentially.

- We observe that, due to CSMA, separating neighboring 802.11b/g radios with one, two or three channels is a worse option than assigning the same channel to them. However, separating neighbor 802.11b/g radios with at least four channels is a better option than assigning the same channel to them. This observation is very valuable for channel assignment algorithms that utilize overlapping channels.

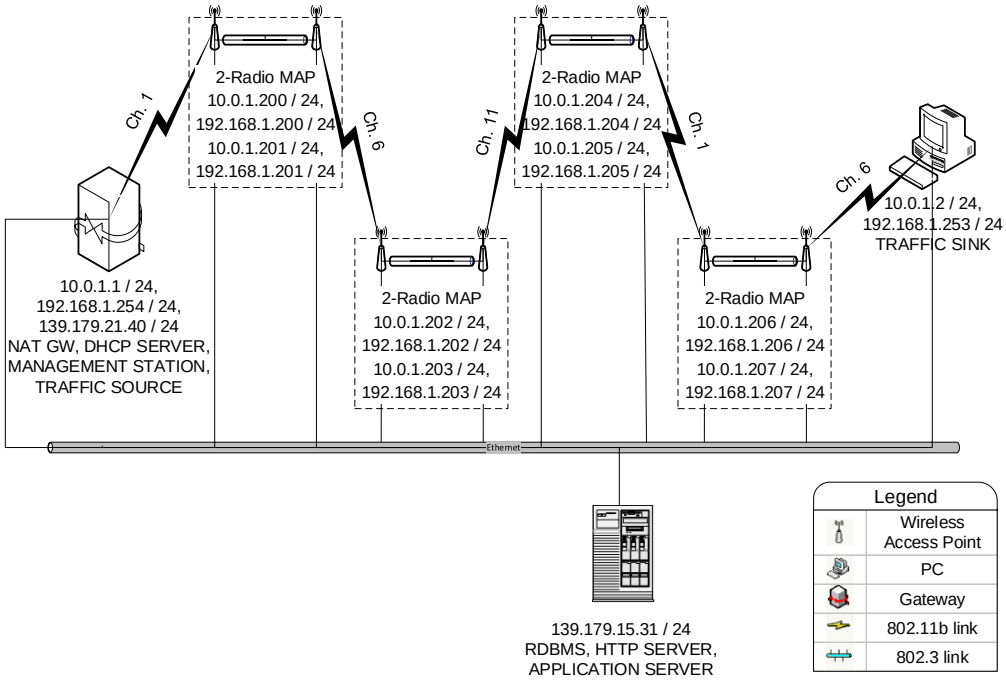


Figure 3.1: BilMesh Logical Topology.

3.2 BilMesh

In this section, we describe our testbed; how we have built, established and configured it. We describe in detail our node hardware and software architecture.

In the Engineering Building of Bilkent University, we have built and deployed an 802.11b/g mesh network, called *BilMesh*, consisting of single-radio and two-radio nodes. We use BilMesh as our testbed for wireless mesh networking research. BilMesh is based on Linksys WAP54G and Linksys WRT54GL 802.11b/g access points running the Whiterussian and Kamikaze distributions of the popular OpenWRT firmware. OpenWRT [56] is a Linux distribution for embedded devices like Wi-Fi access points that provides a fully writable file system with package management. Since BilMesh is based on commodity hardware and open source software, we can easily add new nodes (or remove existing ones) when necessary. Furthermore, since our two-radio nodes are built from conventional single-radio nodes, when desired, we can easily turn our multi-radio nodes into

single-radio ones.

One Linux PC is configured as the mesh network's Internet gateway and DHCP server. During performance measurements, it also acts as the traffic sink. Another Linux PC on the other end of the network is used as the traffic source. As part of BilMesh, we also have a management and monitoring station also running Linux and a server station running MySQL RDBMS, Apache web server and Apache Geronimo J2EE application server [57] on top of Linux.

The Linksys WAP54G [58] is a rather restricted hardware platform for mesh networking with a 200 MHz Broadcom CPU, 2 MB flash memory and 8 MB RAM. It is based on the BCM4318 SoC [59] integrating a CPU and an 802.11b/g interface. The Linksys WRT54GL [60] is a more powerful platform compared with the WAP54G, offering 4 MB flash memory and 16 MB RAM. WRT54GL is based on the Broadcom BCM5352 SoC router [61] which combines a 200 MHz MIPS32 CPU, an 802.11b/g interface and a configurable five port Fast Ethernet switch.

Figure 3.1 shows the logical topology of BilMesh together with the architectural roles of its constituent nodes and Figure 3.2 shows a logical two-radio node. In Figure 3.1, MAP stands for Mesh Access Point, i.e., a wireless router (which we also call as a mesh relay), which can have single or multiple (two) radios. In Figure 3.2, the two physically separate APs are connected via Ethernet to constitute a single two-radio node.

Each node in the testbed (including both of the constituent wireless routers of a two-radio node) is connected to an Ethernet backbone which is used for managing and monitoring the testbed. Using this backbone, channels of the wireless routers can be reliably configured and real-time packet traces can be collected. Also all experiments carried on the testbed can be remotely controlled to prevent any unwanted fluctuations in wireless link conditions caused by moving bodies.

As the routing protocol, OLSR [54] is run on BilMesh. Each constituent router of a two-radio node runs an instance of the OLSR daemon (olsrd [55])

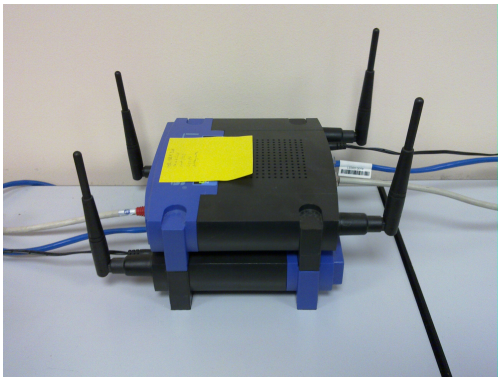


Figure 3.2: A BilMesh two-radio node (Mesh Access Point) consisting of two distinct APs.

which sets up the routing table. Further details on routing software configuration and operation are given in the next section.

3.2.1 Node Configuration

All nodes in BilMesh operate in the 802.11 IBSS mode. In its default configuration on Whiterussian distribution, the 802.3 and the 802.11 interfaces of a WAP54G are bridged together. We break this bridge and configure the wired and wireless interfaces separately, so that packets arriving at the wireless interface can be routed through the wired interface. For maintenance purposes, the nodes can be accessed via their Ethernet interfaces. RTS/CTS is disabled in the network.

Figure 3.3 shows the architecture based on the OpenWRT firmware for a WAP54G node. The wireless interface `eth1` is removed from the bridge `br0` (which contains `eth1` in the default configuration) and is configured to be in the 10.0.1.0/24 network. The VLAN `vlan0` consists of ports 1 and 5 of the programmable switch `et0` and is not tagged. For ease of configuration, VLAN `vlan0` is configured via the default bridge `br0` (from which the wireless interface has been removed) in the 192.168.1.0/24 network. Interface configuration is performed via `init` scripts. Also in these init scripts, the routing information for a specific node can be supplied if static routing is desired, in which case, OLSR daemon should also be disabled. Similarly, Figure 3.4 shows the architecture for a WRT54GL

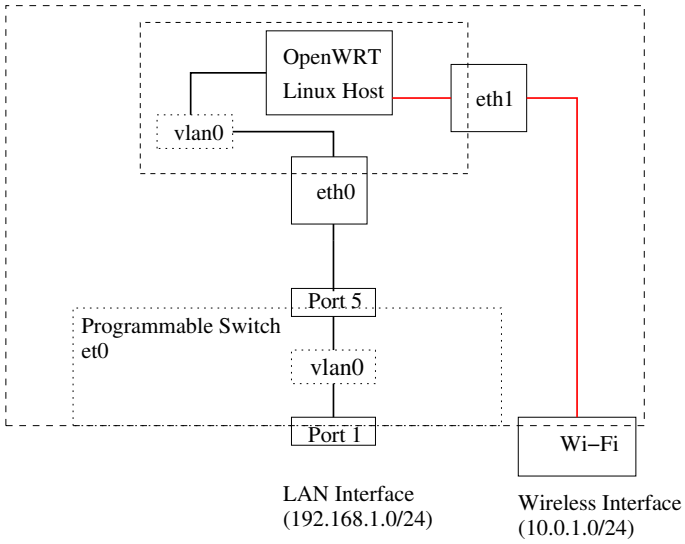


Figure 3.3: OpenWRT based architecture for a WAP54G in BilMesh.

node. Again, the wireless interface has been removed from bridge `br0`.

The routes followed by packets when ad hoc dynamic routing is used can change quite often even in infrastructure meshes like BilMesh since the conditions of the wireless medium change rapidly. If a set of experiments to be carried on BilMesh requires the network packets to follow the same routes, then we disable `olsrd` and use static routing, i.e., the routing table is stored at boot time in the network layer of a node's TCP/IP stack. Also for the static routes to be forced, ICMP Redirect message [62] generation and processing are disabled at the Linux TCP/IP stack. This can be achieved by setting the keys `net/ipv4/conf/all/send_redirects` and `net/ipv4/conf/all/accept_redirects` to 0 in the sysctl preload/configuration file.

3.2.2 Building Two-Radio Nodes

We have built the two-radio nodes out of two WAP54G or two WRT54GL or one WAP54G and one WRT54GL devices. In order to achieve a two-radio node, we connected two WAP54G/WRT54GL devices via their Ethernet interfaces and

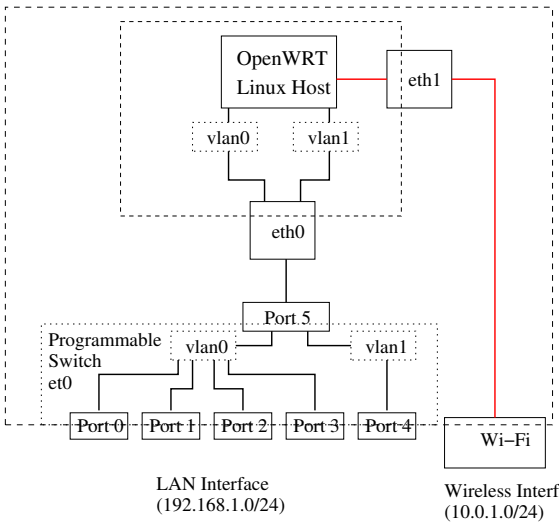


Figure 3.4: OpenWRT based architecture for a WRT54GL in BilMesh.

supplied the necessary routing information to route packets received via the radio interface of a box to the radio interface of the other box in the init scripts. Since packets are routed from one radio to the second radio of this two-radio node via the interconnected 802.3 interfaces, these two radios may be operated on different channels. This effectively gives us a two-radio, two-channel node in which the two radio interfaces can be configured independently of each other. Since the 802.3 link in our setup has a dedicated bandwidth of 100 Mbps, the bottleneck links are the wireless links. Figure 3.5 shows the architecture of a dual radio node which consists of two WAP54G boxes.

Since our two-radio nodes are built using two separate physical boxes, a single instance of the OLSR daemon cannot access both radios of the logical two-radio node. However, in order to be able to route packets between these radios which may be operating on different frequencies (channels), we need to have these radios discover each other with OLSR HELLO messages [54]. Furthermore, the Multipoint Relay (MPR) Selection Sets of each radio must be disseminated to every other radio in the network regardless of the operating channels. To solve these problems we adopted the following approach: on each constituent router of each logical two-radio node, a single instance of the OLSR software is run. OLSR daemon is configured to operate on both the wired (802.3) and the wireless (802.11) interface of the constituent router. In this way, we were able to

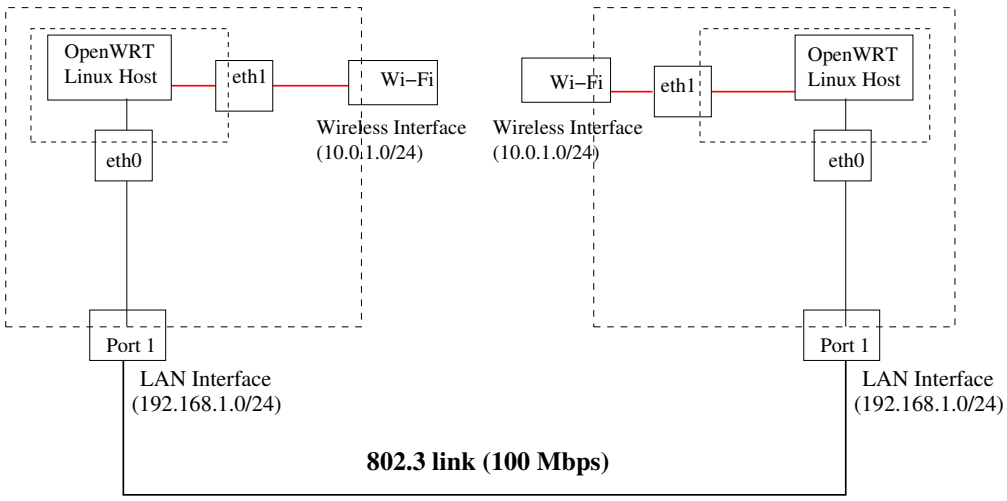


Figure 3.5: A dual radio node comprising two WAP54G hardware.

disseminate network-wide routing information among radios operating on different frequencies, which would not be possible if the daemon operated only on the radio interfaces of the routers. Listing 3.1 contains the related section of the `olsrd` configuration file. The single instance of the daemon is instructed to work on both of the wired (`eth0.0`) and the wireless (`wl0`) interfaces.

```
Interface "wl0"
{
        Ip4Broadcast 255.255.255.255
        HelloInterval 2.0
        HelloValidityTime 40.0
        TcInterval 5.0
        TcValidityTime 100.0
        MidInterval 18.0
        MidValidityTime 324.0
        HnaInterval 18.0
        HnaValidityTime 108.0
}

Interface "eth0.0"
{
        Ip4Broadcast 255.255.255.255
        HelloInterval 2.0
        HelloValidityTime 40.0
        TcInterval 5.0
        TcValidityTime 100.0
        MidInterval 18.0
        MidValidityTime 324.0
        HnaInterval 18.0
        HnaValidityTime 108.0
}
```

Listing 3.1: Part of OLSR Daemon Configuration on a Multi-Radio Node.

Using our WAP54G/WRT54GL single and two-radio nodes together with our desktop PCs (and laptops) as endpoints, we performed extensive experiments on our testbed with different network configurations and scenarios. In the following section, we describe in detail our experimental setups and report the results of our experiments.

Node Id	Local Address	Next Hop To PC 2	Next Hop To PC 1
PC 1	10.0.1.1	10.0.1.200	-
WR 1	10.0.1.200	192.168.1.201	10.0.1.1 (PC 1)
WR 2	10.0.1.201	10.0.1.202	192.168.1.200
WR 3	10.0.1.202	192.168.1.203	10.0.1.201
WR 4	10.0.1.203	10.0.1.204	192.168.1.202
WR 5	10.0.1.204	192.168.1.205	10.0.1.203
WR 6	10.0.1.205	10.0.1.206	192.168.1.204
WR 7	10.0.1.206	192.168.1.207	10.0.1.205
WR 8	10.0.1.207	10.0.1.2 (PC 2)	192.168.1.206
PC 2	10.0.1.2	-	10.0.1.207

Table 3.1: Routing table configurations of the nodes of the five-hop topology (entries for the 802.3 interfaces not shown).

3.3 Experiments

We have conducted experiments on single-hop and multi-hop (up to 5 hops) topologies carrying both UDP and TCP traffic. The TCP and the UDP traffic is generated using the Iperf tool [63] on Linux. For two-hop and three-hop topologies, we have built two-radio, two-channel relay nodes and repeated our experiments to compare the results with their single-radio counterparts. To obtain stable routes for controlled experiments, we used static routing as explained in Section 3.2.1. For each topology, we have also measured RTTs for packets of sizes of 64, 350, 700 and 1470 bytes and we report the jitter values of UDP traffic for each setup. In the discussions that follow, the definition of jitter follows the definition of *Interarrival Jitter* in RFC 3550 [64]. Also for the two-hop multi-radio relay node setup, we investigate the effects of channel separation between the interfaces of the two-radio relay node on network performance.

For multi-hop topologies, each intermediate router forwards a packet it receives to the next router in the chain towards the packet's destination. As an example, the routing tables of the nodes of the five-hop topology are given in Table 3.1.

The following two subsections discuss the single-radio relay node and two-radio relay node setups separately.

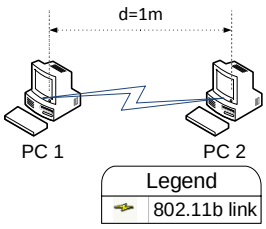


Figure 3.6: Experimental setup for a single-hop network.

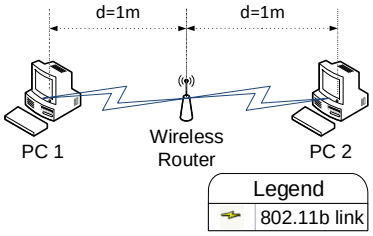


Figure 3.7: Experimental setup for a two-hop network with single radio nodes.

3.3.1 Experiments with Single-Radio Relay Nodes

In order to find the TCP and UDP goodputs achievable on an 802.11b link in our setup, we performed a set of experiments on a one-hop topology. Since the sender and the receiver are only one-hop away from each other, there is no interference from consecutive hops of the stream and signals belonging to other co-located wireless networks constitute the primary source of interference. Figure 3.6 shows the setup for the goodput measurement experiments on a single-hop topology. `PC 1` and `PC 2` are connected together via an 802.11b link on channel 1 at 11 Mbps in IBSS mode. `PC 1` generates UDP traffic with a demand of 11 Mbps targeted at `PC 2`. 15 goodput measurements were made with this setup and the average goodput was found to be 6896 Kbps. Another set of 15 goodput measurements were performed where `PC 1` generates TCP traffic targeted at `PC 2`, and the average goodput was found to be 5438 Kbps. The average jitter for UDP packets was 0.45 ms for this setup.

Figure 3.7 shows the setup for the goodput measurement experiments involving single radio nodes in a two-hop topology. The box labeled as `Wireless Router` (called `WR` from now on) is a WRT54GL. `PC 1`, `PC 2` and `WR` form an 802.11 IBSS (Independent Basic Service Set) on channel 1. All links are 802.11b links at 11 Mbps. Nodes are placed purposefully close to one another (each separated by 1 m) to increase the *intra-flow interference*, which refers to the interference on a link of a flow caused by the subsequent links used by the same flow. `PC 1` generates UDP traffic with a demand of 11 Mbps targeted at `PC 2` but instead

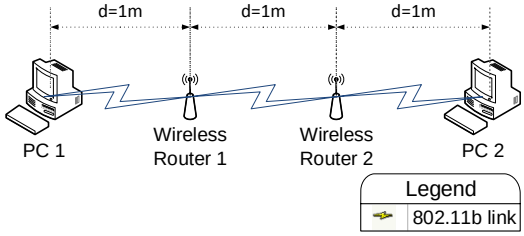


Figure 3.8: Experimental setup for a three-hop network with single radio nodes.

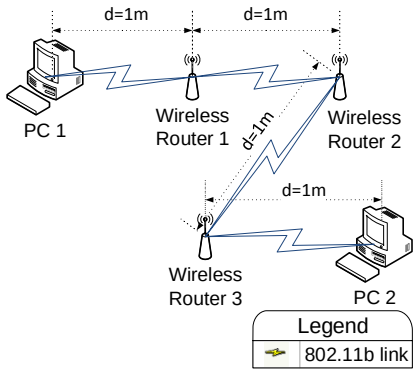


Figure 3.9: Experimental setup for a four-hop network with single radio nodes.

of directly sending this traffic to `PC 2`, `PC 1` asks the `WR` to relay this traffic to its destination. 15 measurements were made with this setup and the average goodput was found to be 3377 Kbps. The average goodput for TCP traffic from `PC 1` to `PC 2` was 2722 Kbps out of 15 measurements and the jitter was found to be 1.67 ms.

For the three-hop topology shown in Figure 3.8, where `PC 1` is the traffic source and `PC 2` is the destination and packets are relayed over `WR 1` and `WR 2`, the average goodput for UDP traffic was found to be 2275 Kbps out of 15 measurements and the average TCP goodput was found to be 1831 Kbps out of 15 measurements. The average jitter for this topology turned out to be 3.55 ms.

For the four-hop topology shown in Figure 3.9, where `PC 1` is the traffic source and `PC 2` is the destination and packets are relayed over `WR 1`, `WR 2` and `WR 3`, the average goodput for UDP traffic was found to be 1570 Kbps out of 15 measurements and the average TCP goodput was found to be 1258 Kbps out of 15 measurements. The average jitter for this topology turned out to be 6.83 ms.

For the five-hop topology shown in Figure 3.10, where `PC 1` is the traffic source and `PC 2` is the destination and packets are relayed over `WR 1`, `WR 2`, `WR 3` and `WR 4`, the average goodput for UDP traffic was found to be 893 Kbps out of 15 measurements and the average TCP goodput was found to be 900 Kbps out of

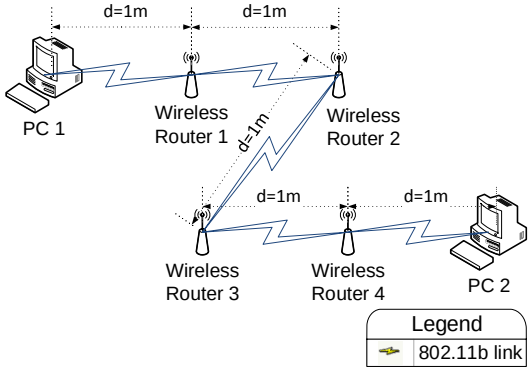


Figure 3.10: Experimental setup for a five-hop network with single radio nodes.

Hop Count	UDP Goodput (Kbps)	TCP Goodput (Kbps)	RTT (ms)	UDP Jitter (ms)
1	6896	5438	3.95	0.45
2	3377	2722	7.54	1.67
3	2275	1831	11.23	3.55
4	1570	1258	14.79	6.83
5	893	900	18.3	13.01

Table 3.2: Averages of the measurements for experiments with single radio relay nodes. RTT averages reported here are for 1470 bytes packets.

15 measurements. The average jitter for this topology turned out to be 13.01 ms.

Table 3.2 summarizes the averages of the results of the measurements obtained on these 1-5 hop topologies with single radio relay nodes. As it can be seen from the experiment results, as the hop count increases both the achievable TCP and UDP goodputs decrease. For smaller hop counts, due to TCP's acknowledgements and congestion and flow control mechanisms, one can achieve larger goodput by using UDP at the transport layer. The interesting fact observed here is that as the hop count reaches 5-hops, TCP can achieve larger goodput than UDP. The UDP source, lacking any transport layer feedback from the subsequent hops, sends as much traffic as CSMA/CA MAC allows. The amount of traffic a UDP source can offer is a function of solely the capacity of the first link in a multi-hop flow (assuming the application always has packets to send as soon as a packet

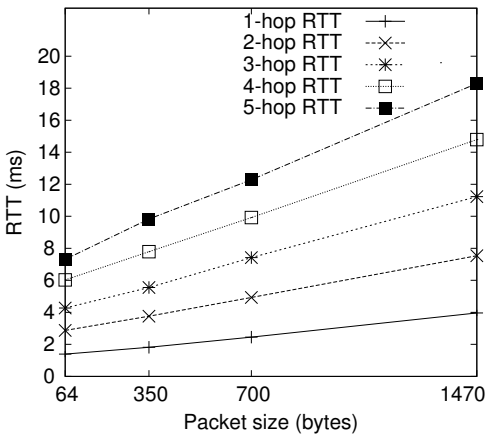


Figure 3.11: RTT measurements for varying sizes of ICMP payloads.

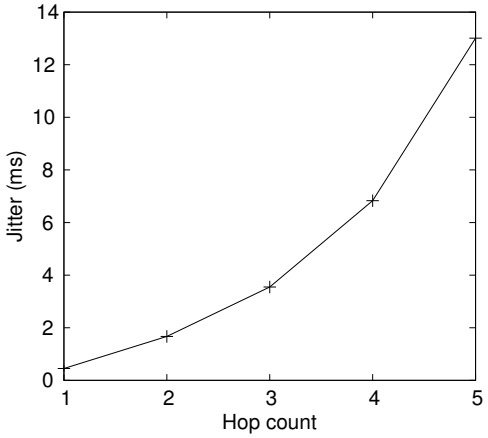


Figure 3.12: Jitter values for UDP packets on 1-5 hop topologies using single-radio relay nodes.

is successfully delivered to the transport layer). However, a TCP source receives transport layer feedback from the traffic destination and throttles itself by means of flow and congestion control mechanisms. As more and more hops are added to a flow, since the links (of the hops) are spatially separated, the capacity of the first link does not change and the UDP source generates packets in a greedy way, that have no chance to reach their destination. However, a TCP sender expects acknowledgement from its receiver and this prevents it from generating unnecessarily large number of packets that would be dropped in intermediate links with high probability. After 4 hops, as TCP's self-throttling mechanisms mitigate congestion among the hops of the flow (*intra-flow congestion*), TCP begins to perform better than UDP in terms of goodput.

For each of these topologies, RTTs were measured with ping packets of 56 (default payload size in iputils ping), 342, 692, 1462 byte payloads. 1462 bytes of ICMP payload corresponds to 1470 bytes of ICMP message together with the 8 byte ICMP header, which in turn is the datagram size used in UDP goodput measurements. Figure 3.11 summarizes the RTT measurements. For all packet sizes, RTT increases almost linearly with respect to increasing hop count and the rate of increase of RTT with respect to hop count increases as the packets grow in size.

Figure 3.12 plots the UDP jitter values on these topologies for datagrams of 1470 bytes. As seen in Figure 3.12, the jitter for UDP packets increase almost exponentially with respect to increasing hop count.

In order to observe the effects of offered traffic volume on the UDP flow goodput, packet drop rates and the jitter, we performed other sets of experiments with the single-radio nodes for 1, 2, 3, 4, 5, 6 and 7 hop cases. In these sets of experiments, the physical link rates are kept constant at 11 Mbps but the offered UDP traffic volume at the source is varied (whereas in the previous sets of experiments, it was kept constant also at 11 Mbps). Also, aluminium foiled panels were used between the hops to decrease the interference range of the transmitter radios. Figures 3.13, 3.14 and 3.15 show the averages of the results for these experiments. The offered traffic load is varied from 1 Mbps to 11 Mbps in increments of 2 Mbps and for each offered load in each topology, a total of 15 experiments were performed. As expected, in Figure 3.13 as the offered load increases, the application level goodput first increases and then saturates at about 1007 Kbps for 7-hops, and at 5447 Kbps for 1-hop. The general trend in Figure 3.13 is that at a given offered load, goodput decreases as the hop count increases since the contention among the links increases. The effects of the aluminium foiled panels are also clearly visible for 2, 3 and 4 hops. If these panels were not used to mitigate inter-hop interference, then one would expect an average goodput of 2750, 1833.3 and 1375 Kbps at maximum for 2, 3 and 4 hops respectively, since the single hop average goodput is below 5500 Kbps. To some extent, these panels have been able to mitigate inter-hop interference. Also as expected, Figure 3.14 shows that at a given offered load, packet drop ratio increases with hop count which is due to increasing intra-flow interference. The same trend also exists for the jitter measurements, however, with some irregularities as it is displayed in Figure 3.15. Jitter values rise as high as 37 ms for the 7-hop topology. The increase in jitter values when going from an offered load of 1 Mbps to 3 Mbps becomes more significant as the hop count increases.

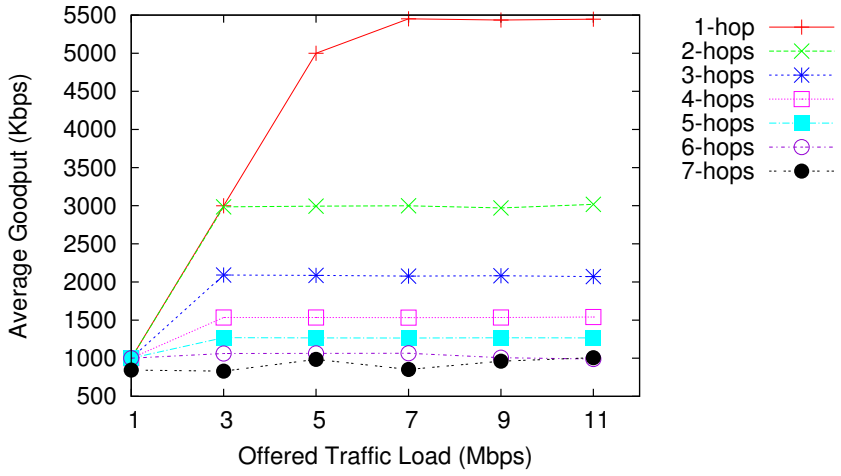


Figure 3.13: Average goodput values for various offered traffic volume for single-radio topologies with 1-7 hops.

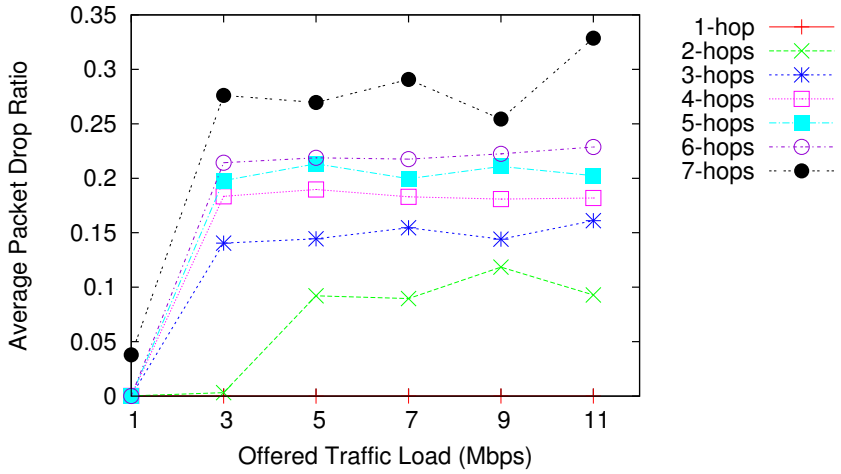


Figure 3.14: Average packet drop ratios for various offered traffic volume for single-radio topologies with 1-7 hops.

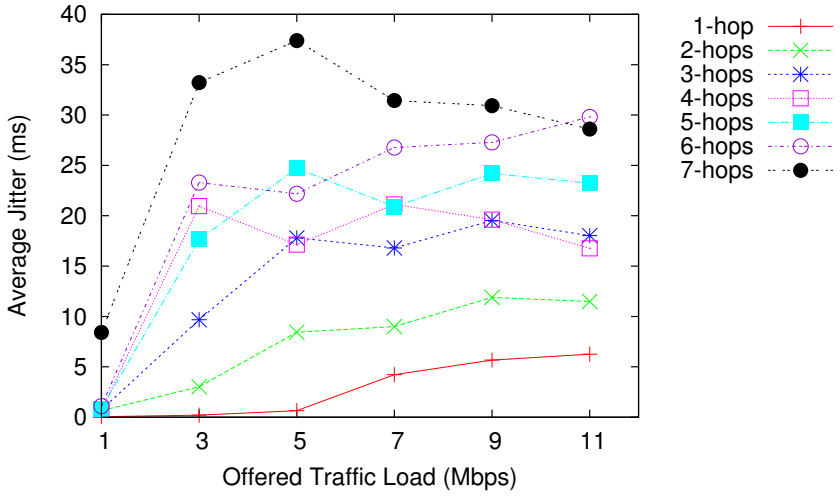


Figure 3.15: Average jitter values for various offered traffic volume for single-radio topologies with 1-7 hops.

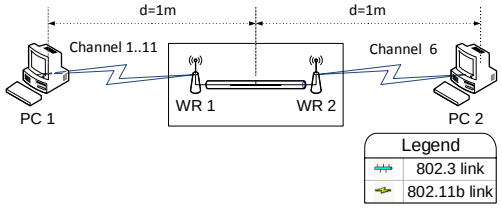


Figure 3.16: Experimental setup involving a two-radio relay node in a two-hop topology.

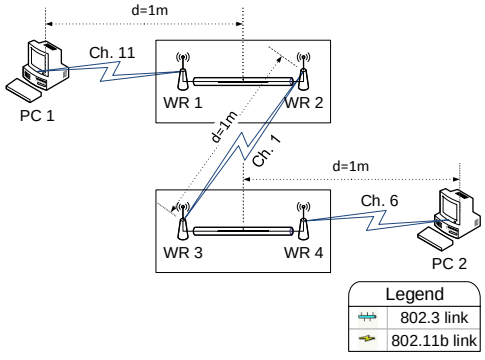


Figure 3.17: Three (wireless) hop setup involving two two-radio relay nodes.

3.3.2 Experiments with Two-radio Relay Nodes

In order to test the viability of using overlapping channels in a multi-radio node setting, we conducted a set of goodput measurement experiments, using Iperf [63] as our traffic generator. Our aim in performing these experiments is to quantify by measurement, the amount of application level performance degradation due to using overlapping channels on a multi-radio relay node. Figure 3.16 shows the setup for goodput measurement experiments involving a two-radio relay node in a two (wireless) hop topology. To obtain a two-radio relay node, the two WRT54GL single-radio wireless routers labeled as `WR 1` and `WR 2` are interconnected via Ethernet and as explained in subsection 3.2.1, each wireless router relays a packet it receives from its radio interface to the other wireless router through the Ethernet connection which then transmits the packet via its radio interface. In this setup, the channel on which the radio of `WR 1` operates is changed from 1 through 11, whereas the channel on which `WR 2` operates is kept constant at 6. `PC 1` connects to `WR 1` in 802.11 IBSS mode and hence the channel on which the radio of `PC 1` operates is also varied accordingly. `PC 2` connects to `WR 2` and hence the radio of `PC 2` is operated on channel 6. `PC 1` generates UDP traffic (with a demand of 11 Mbps) targeted at `PC 2` and the system of wireless routers consisting of `WR 1` and `WR 2` acts as a *two-radio relay node* to carry this traffic. All 802.11 radios in this setup operate in the IBSS mode at 11 Mbps. For each channel configuration, 6 goodput measurements are performed with a total of 66 measurements. Each measurement lasts 10 seconds. Figure 3.18 depicts the normalized average goodput values obtained through these measurements. The averages are normalized with respect to the average goodput obtained when `WR 1` is at the same channel as `WR 2` (channel 6). The maximum average UDP goodput is obtained when `WR 1` is at channel 11 and `WR 2` is at channel 6. As it can be seen from Figure 3.18, when the separation between the two channels of the relay node is 4, we have a goodput gain of at least 113% compared with a channel separation of less than 4.

Also another interesting observation is that when `WR 1` is set to operate on channels 3, 4, 5, 6, 7, 8 or 9, we have a local maximum at channel 6 which is the channel occupied by `WR 2`. This can be attributed to CSMA. Carrier sensing is

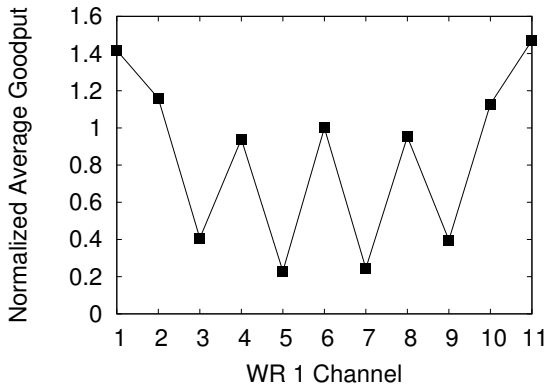


Figure 3.18: Normalized average goodput measurements in the setup involving a two-radio relay node.

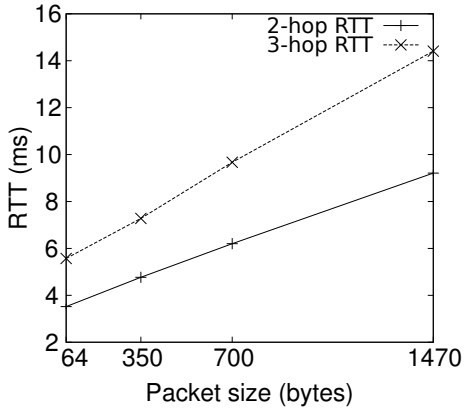


Figure 3.19: RTT measurements for multi-radio relay setups with varying sizes of ICMP payloads.

more effective when both radios are on the same channel compared with the cases when two radios operate on channels that are 1-3 channels away, reducing packet losses and increasing the goodput. This observation might be valuable for channel assignment tasks involving multi-radio nodes. If a channel assignment algorithm being designed allows overlapping channels to be assigned to neighboring radios, it is better to assign the same frequency to these radios instead of assigning channels that are one, two or three channels away.

Figure 3.17 shows the setup consisting of two two-radio relay nodes in a three (wireless) hop topology. `WR 1` and `WR 2` are interconnected via Ethernet and form a two-radio relay node as explained in Section 3.2.1. Similarly, `WR 3` and `WR 4` are interconnected via Ethernet to form another two-radio relay node. In this topology, `PC 1` communicates with `PC 2` via 5 hops two of which are 802.3 links (hence there are 3 wireless hops). In this setup, `WR 1` operates on channel 11, `WR 2` operates on channel 1, `WR 3` operates on channel 1 (so that there exists an 802.11b link between `WR 2` and `WR 3`), `WR 4` operates on channel 6. `PC 1` and `PC 2` operate on channels 11 and 6, respectively. Hence, all of the three wireless links are operated on orthogonal channels. As explained previously, nodes are placed close to one another (separated by 1 m) to increase intra-flow interference. The average UDP goodput is measured to be 5401 Kbps and the average TCP goodput is found to be 3055 Kbps for this setup. Jitter is observed to be 2.05 ms. When these

results are compared with their counterparts of the single radio three wireless hops case in Table 3.2, it can be observed that there is a goodput improvement of about 237% for UDP traffic and about 167% for TCP traffic for 3-hop topologies. The jitter values for the multi-channel case decrease less sharply, by about 42%, compared with the single channel three hops case. We may conclude from these results that the goodput gains for UDP and TCP traffic in the multi-channel case are higher than the jitter gains. This can be attributed to the additional queues introduced with the 802.3 links in the multi-channel setting.

For both multi-radio setups, RTTs were measured for ping packets of 64, 350, 700, 1470 bytes (including ICMP headers). Figure 3.19 summarizes the averages for these RTT measurements for the two multi-radio relay node topologies discussed. When compared with the performances of their single-radio counterparts depicted in Figure 3.11, it can be seen that RTT values are higher in the multi-radio case. The difference comes from the additional 802.3 links and the additional store-and-forward delays introduced in our multi-radio setup. For a two-hop topology, in the single-radio setting, there is only one intermediate (wireless) router relaying the traffic, whereas in the multi-radio setting of our setup, there are two such routers. We should note here, that the ping packets used for measuring the average RTT for a given ping packet size are sent with a separation of 1 s and they are not flooded. Since these ping packets are not flooded (i.e. are not sent back to back), they do not experience intra-flow interference. If these ping packets were flooded, then the multi-radio setup would have an advantage over the single-radio setup because of the mitigation of the intra-flow interference in the multi-radio multi-channel setting.

In order to observe the effects of offered traffic volume on the application goodput in the multi-radio case, we repeated the previously described set of experiments with multi-radio relay nodes for two and three hop topologies. Again, the physical link rates are kept constant at 11 Mbps but the offered UDP traffic volume at the source is varied from 1 Mbps to 11 Mbps in increments of 2 Mbps. For the two (wireless) hop topology, we employed channels 1 and 6, and for the three (wireless) hop topology, we employed channels 1, 6 and 11. The setup for the 2-hop topology is similar to that of Figure 3.16 with the only difference that

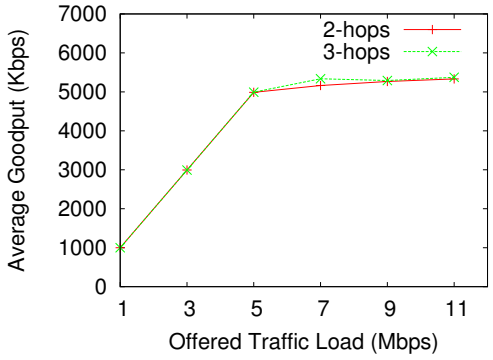


Figure 3.20: Average goodput values as offered traffic volume changes for 2-hop and 3-hop two-radio topologies.

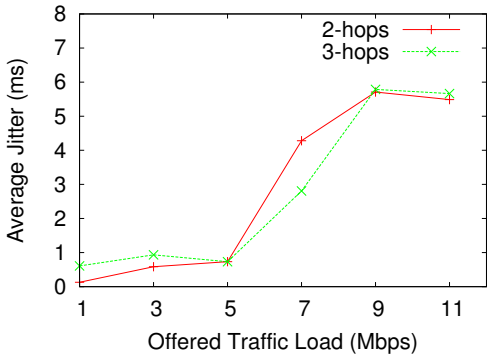
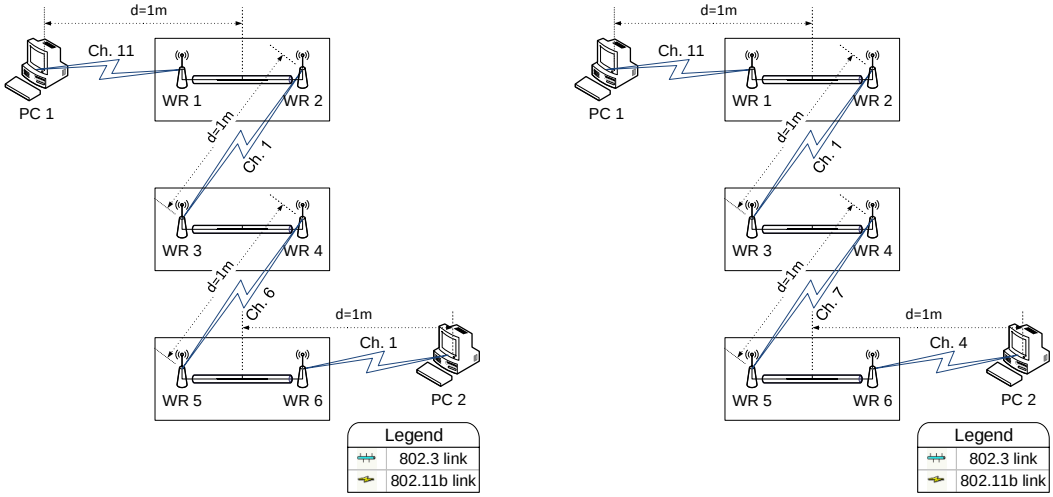


Figure 3.21: Average jitter values as offered traffic volume changes for 2-hop and 3-hop two-radio topologies.

the link between `PC 1` and `WR 1` operates on channel 1. The setup for the 3-hop topology is identical to Figure 3.17. Figures 3.20 and 3.21 show the averages of the results for these experiments. As it can be seen from Figure 3.20, when non-overlapping channels are used, the maximum achievable goodput does not differ significantly between two and three hops. When the average goodputs reported in Figure 3.20 are compared with their single-radio node counterparts of Figure 3.13, we can see that, due to more parallel transmissions in the multi-radio case, maximum average goodputs for 2 and 3 hops have increased up to 170% and 237% respectively. We also observe that, in the multi-radio node setup, the difference in the maximum goodputs of 2 and 3 hop flows have decreased significantly because the flows do not experience intra-flow interference even for 3 hops. Figure 3.21 shows that the jitter values are below 1 ms for offered loads of 1, 3 and 5 Mbps. But as the offered load rises above 5 Mbps, jitter increases rapidly.

We also experimented with a 4-hop multi-radio topology to assess if goodput can be improved by using distinct overlapping channels instead of repeating non-overlapping channels on different links. Since the number of non-overlapping channels is 3 in IEEE 802.11b/g, we need at least 4 wireless hops to investigate this fundamental question. We used the two topologies depicted in Figure 3.22 for these sets of experiments. In Figure 3.22(a), only non-overlapping channels $1, 6, 11$ are used but since we have 4 wireless hops, one of these channels



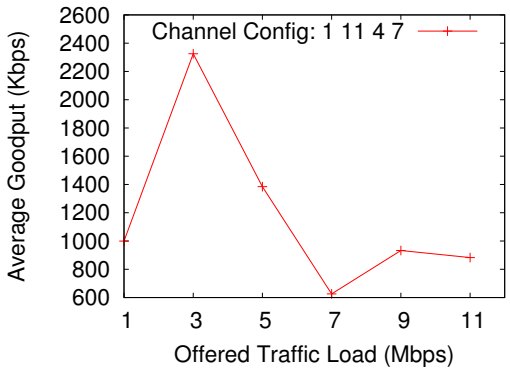
(a) Using Only Non-overlapping Channels.

(b) Allowing Overlapping Channels.

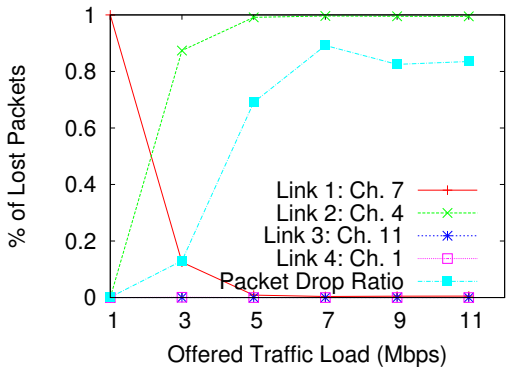
Figure 3.22: Motivational Example: Is using channels $1, 6, 11$ solely and repeating channels when needed better, or is allowing overlapping channels better?

has to be repeated (e.g., channel 1 is used on links 2 and 4). However, in Figure 3.22(b), overlapping channels $1, 4, 7, 11$ are used and no channel is repeated on subsequent links. Our aim is to investigate whether allowing overlapping channels to be used improves performance over repeating channels on subsequent links.

We performed several goodput measurement experiments with the 4-hop topology depicted in Figure 3.22 using various permutations of channels. We report the results of some selected scenarios here. In each of these scenarios, the scenario name consists of the channel numbers of the links from the traffic sink to the traffic source in order. The channel configuration is also listed in the same order. For instance, if the scenario name (or channel configuration) is $4, 7, 1, 11$, then in the related experiment from the traffic source to the traffic destination, the 1^{st} link operates on channel 11, the 2^{nd} link operates on channel 1, the 3^{rd} link operates on channel 7 and the last link operates on channel 4. For these sets of experiments, all links are 802.11b links operating at 11 Mbps and the transmit powers for all of the transmitters are fixed at 17 dBm (about 50 mW). Also in order to observe link-level packet losses, using pcap library, we collected packet



(a) Average Goodput vs. Offered Load for 4-hop Scenario $1, 11, 4, 7$.



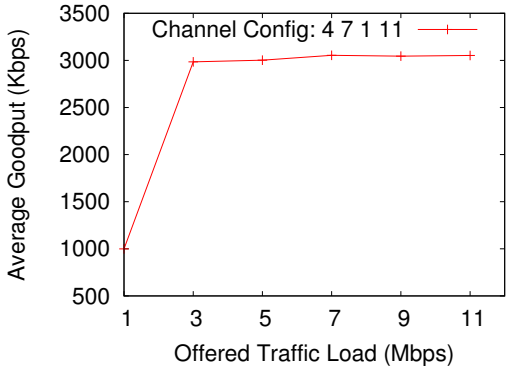
(b) Link Loss Percentages for 4-hop Scenario $1, 11, 4, 7$.

Figure 3.23: 4-hop Scenario with Channels $1, 11, 4, 7$.

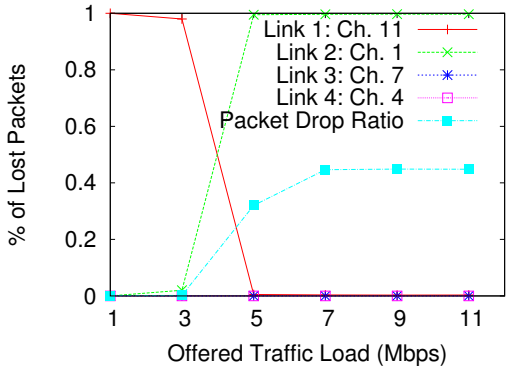
traces on wireless routers labeled as `WR 1, WR 3, WR 5` and on the traffic destination (`PC 2`) using a pcap snap length of 70 bytes. A snap length of 70 bytes is enough to capture the application level packet header used by Iperf. Iperf assigns a *packet identifier* incremented by 1 to each packet it generates and puts it in the application layer packet header. Also at the end of the session, Iperf traffic source reports to the traffic sink the total number of packets generated during the session. Doing a post analysis on the packet traces after the experiment is completed, we were able to identify the packet loss rate on each of the four wireless links individually. Each experiment is repeated 10 times.

In Figures 3.23(a), 3.24(a) and 3.25(a), we report the averages of the goodput measurements with respect to increasing offered traffic load. And in Figures 3.23(b), 3.24(b) and 3.25(b), we report the averages of the percentages of lost packets on individual links. In Figures 3.23(b), 3.24(b) and 3.25(b), "Packet Drop Ratio" represents the average overall packet loss ratio, which is the ratio of the total number of lost packets (on Link 1, 2, 3 or 4) to the total number of packets sent by the traffic source.

In our experiments, we have observed that if we stick to only non-overlapping 802.11b channels, we obtain almost the same goodput for different permutations of channels *as long as channel repetition on neighboring links* (links incident on a common node) is not allowed. But if we employ overlapping channels, then the

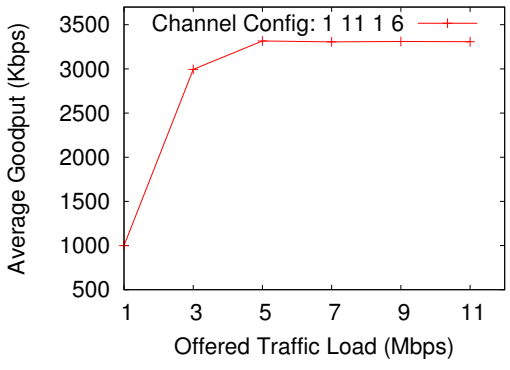


(a) Average Goodput vs. Offered Load for 4-hop Scenario $4, 7, 1, 11$.

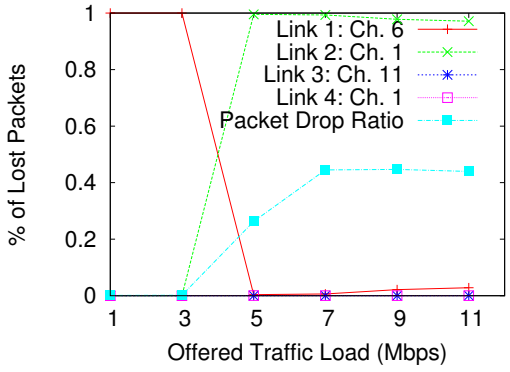


(b) Link Loss Percentages for 4-hop Scenario $4, 7, 1, 11$.

Figure 3.24: 4-hop Scenario with Channels $4, 7, 1, 11$.



(a) Average Goodput vs. Offered Load for 4-hop Scenario $1, 11, 1, 6$.



(b) Link Loss Percentages for 4-hop Scenario $1, 11, 1, 6$.

Figure 3.25: 4-hop Scenario with Channels $1, 11, 1, 6$.

permutation of channels chosen has a more profound impact on goodput.

Figures 3.23 and 3.24 reveal an interesting fact. Although the channel subset used for the 4 hops of the network is exactly the same in these scenarios, maximum achievable goodput differs by 31% (2325.3 Kbps vs. 3053.8 Kbps). Separating the first two links by three channels performs considerably worse than separating the last two links by three channels. The reason for this phenomenon is that in our setup, links operating three channels away from each other severely interfere with each other because of being spatially close. If the three channel separation is used between the first and the second links on which more packets are carried compared with the third and the fourth links, more packets are lost due to collisions. However, if the three channel separation is used between the last two links that carry less traffic due to the thinning effect, and interference between the more loaded first two links is kept relatively low (i.e., by employing non-overlapping channels), relatively less number of packets are lost.

Another observation that follows from these figures is that having *higher* inter-link intra-flow interference at the *beginning* of a flow (e.g., interference between the first two links of the *same* flow) makes the flow *less stable* with respect to increasing load. If we consider Figure 3.23, increasing the number of packets on the first link by increasing the offered load, decreases goodput up to 73% (when the offered load is 7 Mbps). However, if the first two links operate on non-overlapping channels and do not interfere with each other as in the scenario given in Figure 3.24, the flow is much more stable with respect to increasing offered load. As it can be deduced from Figures 3.24 and 3.25, on a linear topology, repeating a channel on non-consequent (two or more hops away) links is a better choice than using overlapping channels in consequent links when goodput is concerned. Again in Figure 3.25, we observe that the obtained goodput as offered load increases is more stable when compared with that of Figure 3.23. In Figure 3.25(b), we observe that for offered loads of 1 and 3 Mbps, nearly 100% of the packet losses occur on the first link but the overall packet loss ratio is almost 0%. But as the offered load is increased beyond 3 Mbps, exceeding the capacity of the path, the overall packet loss ratio jumps to over 40% and almost all of the losses occur on the second link. The situation is similar for the scenario $4, 7, 1, 11$ as it can be

observed in Figure 3.24(b). On both of these scenarios, the first and second links are operated on non-overlapping channels. If overlapping channels are used on the first and second links as in Figure 3.23(b), at an offered load of 3 Mbps, the overall packet loss ratio rises to 13% and like the previously mentioned scenarios, most of the packet losses (about 87%) are on the second link. However, an offered load of 3 Mbps is above the path capacity for this scenario.

When we consider link-level packet losses, it can be seen from Figures 3.23(b), 3.24(b) and 3.25(b) that the second link is the most vulnerable link in our experiments under heavy traffic load. This is because packet losses occur at the transmit queues of the nodes rather than at the links themselves. Since the first link's transmit queue is at the PC and is larger than the subsequent router transmit queues and since the number of packets making it to the 3^{rd} and the 4^{th} links' transmit queues is substantially smaller, more packets are dropped at the 2^{nd} link's transmit queue. This is in accordance with our observation stated above, that it is more important to protect the head of a flow from interference (intra-flow or external) than to protect the tail. When assigning channels to radios or when *making routing decisions*, this fact must be taken into account.

3.4 Summary

In this chapter, we report on an indoor 802.11b/g mesh networking testbed we established at Bilkent University as part of this thesis and provide experimental evaluation results on multi-hop topologies for TCP and UDP traffic. The achievable goodput quickly drops as the hop count increases when operating on a single channel, but employing multi-radio, multi-channel nodes as the intermediary relaying nodes can provide up to 192% improvement in UDP goodput and up to 176% improvement in TCP goodput in a two-hop topology. The UDP goodput improvement reaches 237% when the flow is three hops long. TCP is more sensitive to the increased packet loss rate and increased RTTs as the hop count increases. With the multi-radio architecture used in our experimental setups, RTTs in multi-hop topologies where packets are relayed by multi-radio nodes

are longer and RTTs grow faster as hop count increases compared with the case where packets are relayed by single-radio nodes. This is due to the additional processing performed by the access points constituting the multi-radio relay node, when routing the packets from the receiving wireless interface of one access point to the transmitting wireless interface of the *other* access point via the 802.3 link.

Despite this adverse effect in our multi-radio architecture, the additional channel capacity obtained by making use of multiple physical radio interfaces results in improvements of achievable goodput up to 167%. Another interesting result is that when utilizing overlapping 802.11b channels for multi-radio nodes, one has to take special care to separate the channels assigned to the radio interfaces appropriately. This is because separating the 802.11b radio interfaces with 1, 2 or 3 channels (corresponding to central frequency separations of 5, 10 and 15 MHz respectively) may severely degrade the achievable performance compared to the case in which the same channel is assigned to the interfaces. This is due to CSMA's incapability to properly coordinate contenders which are separated by 1, 2 or 3 channels, increasing the rate of packet collisions. On the other hand, a separation of 4 channels, which implies the assignment of slightly overlapping channels in the context of 802.11b, achieves goodput improvements of up to 189% for UDP traffic in a two-hop topology when compared to the single-radio case. As mentioned above, using non-overlapping channels achieves even higher goodput improvements. According to the results reported, operating the radio interfaces of a multi-radio relay node on the same channel effectively turns it into a single-radio relay node from the perspective of network performance for UDP traffic without providing any advantage of using multi-radio nodes.

Chapter 4

A Novel Measurement-based Approach for Modeling and Computing Interference Factors for Wireless Channels

Wireless communication technologies divide their available spectrum into predefined channels. Some wireless technologies, such as the IEEE 802.11b/g, define their channels in such a way that adjacent channels share the spectrum. When two distinct channels share some part of their spectrum, simultaneous transmissions on these channels cause what is classified as the *adjacent channel interference* [2,46,65–67]. For problems that consider adjacent channel interference, such as the channel assignment problem for multi-channel wireless mesh networks, we need a model that quantitatively describes adjacent channel interference. The interference factor is a concept defined to quantify the amount of the overlap, hence the interference, between two wireless channels. Analytical and experimental methods have been proposed in the literature to define the interference factors

among channels of a wireless technology. In this chapter[1], we propose a physical-layer-measurement-based, technology-independent and generic approach that is capable of determining interference factors between the channels of a wireless technology and also between the channels of two different wireless technologies, such as IEEE 802.11 and IEEE 802.15.4. We also report our measurement results for interference factors among 802.11b DSSS channels and between 802.15.4 and 802.11b channels. Our results show that our approach is practical, accurate and generic enough to compute the interference factors of radio channels belonging to various wireless communication technologies.

In Section 4.1, we discuss our motivations in our study of the I-factor concept and list our major contributions regarding I-factor. In Section 4.2, we introduce the concept of the I-factor in more detail. In Section 4.3, we present and discuss our novel methods. In Section 4.4, we report our measurement results for the I-factors between 802.11b DSSS channels and between 802.15.4 and 802.11b channels. Finally, in Section 4.5, we conclude the chapter.

4.1 Introduction

Interference factor (I-factor) [25] is used to model the amount of interference between two channels of a wireless communication standard, such as the IEEE 802.11 family of standards or the IEEE 802.15.4 standard. Spectral bands are allocated to wireless communication standards by regulation and standardization bodies such as the US Federal Communications Commission (FCC) or the European Telecommunications Standards Institute (ETSI). These standards further divide the band allocated to them into channels. Each standard has its own channel definition, some of which are compatible and some of which are not. For instance, the IEEE 802.11g standard is compatible with the 2.4 GHz DSSS channel definitions of the 802.11b standard for supporting legacy devices, whereas the

[1]The material presented in this chapter has been previously published by SpringerOpen at the EURASIP Journal on Wireless Communications and Networking, and is available as an Open Access article [68]. The figures and tables in this chapter are reprinted with permission from the publisher.

channel definitions of the 802.11g and IEEE 802.15.4 standards are not compatible with each other. Furthermore, a standard may define more than one channel structure if it employs multiple PHY service specifications. As an example, the IEEE 802.11 standard defines different channel structures for the Frequency-Hopping Spread Spectrum (FHSS) PHY and the Direct-Sequence Spread Spectrum (DSSS) PHY services [22].

The definition of the channels of a wireless communication standard comprises the center frequencies and the bandwidths of the channels. For instance, the IEEE 802.11b and IEEE 802.11g standards use the same channel structure for the DSSS PHY. The 802.11b/g DSSS channels are in the 2.4 GHz Industrial, Scientific and Medical (ISM) band. The center frequency of the first 802.11b/g DSSS channel (channel id 1) is 2412 MHz. The center frequencies of consecutive channels are separated by 5 MHz and the bandwidth of each channel is 22 MHz. The standard defines 14 channels, of which the first 11 are supported in the FCC domain.

A receiver radio may be exposed to interference energy from a transmitter (interferer) radio sending packets on the same channel as itself. This is called co-channel interference. To further complicate the situation, the interfering radio might be operating on a different (but adjacent or overlapping) channel as the channel the receiver under discussion is using. Some wireless communication standards (for example, IEEE 802.11a) define their channel structures in such a way that it is impossible for a radio to receive any signal power from a transmitter not tuned to the same channel as itself. However, for other standards (such as the 802.11b/g DSSS PHY), it is possible for a receiver operating on channel i to receive interference power from a transmitter operating on one of the channels $i, i \pm 1, i \pm 2, i \pm 3$, or $i \pm 4$. This occurs because of how DSSS channels as well as the transmit spectrum mask (TSM) are defined in 802.11b/g. The DSSS TSM [22] allows a transmission bandwidth of 22 MHz centered around the channel center frequency. Therefore, two transmitters tuned to channels i and j *share* the wireless medium (a common frequency range) as long as i and j's center frequencies are separated by less than 22 MHz. Because the center frequencies of consecutive channels are 5 MHz apart from each other, this translates into a channel separation, $|i - j|$, of fewer than $\lceil 22/5 \rceil = 5$ channels.

The term *overlapping* channels [25] (or *non-orthogonal* channels) is used to describe a relation between at least two channels that share a frequency range. When we consider the 802.11 DSSS PHY specification, two channels i and j are overlapping channels if and only if $|i - j| < 5$. If $|i - j| \geq 5$, channels i and j are called *non-overlapping* or *orthogonal* channels.

The interference factor quantitatively defines the amount of overlap and interference between two wireless channels. In this chapter, we propose a novel and practical approach that can be used to compute the interference factor values between not only the channels of a single standard but also between the channels of two different standards sharing the same spectrum. Our approach is based on physical-layer measurements and has the advantage of being practically applicable to various wireless communication standards. Existing analytical or measurement-based approaches lack this important property because of their dependence on a specific wireless standard. Our main contributions are:

- To the best of our knowledge, the physical-layer-measurement-based methods we propose are the first in the literature that are generic enough to model the interference between channels of any two wireless communication technologies, such as 802.11 and 802.15.4. They can also be used to obtain the I-factor values between channels of the same wireless technology, such as 802.11 channels.

- These methods are also capable of modeling interference on wireless communication devices caused by non-communication devices such as microwave ovens.

- Using the proposed methods and a 2.4 GHz spectrum analyzer, we give our measurement results for the proposed I-factors between 802.11b DSSS channels and we compare our results with those of other analytical and measurement-based I-factor models in the literature. We also report our measurement results on the interference from an 802.15.4 (ZigBee) transmitter on an 802.11b receiver.

4.2 Interference Factor

The concept of an interference factor [25] is used as a model of how much interference power will leak from adjacent channels. An I-factor is commonly defined as a value in the interval $[0, 1]$, where 0 means no interference, and 1 means maximum interference (when the interferer's frequency band intersects maximally with the transmitter's frequency band). Ideally, a transmitter operating on a non-overlapping channel with respect to a receiver generates no interference on the receiver, therefore the I-factor for two non-overlapping channels is 0. When the interferer and receiver radios are operating on the same channel, 100% of the interferer signal power at the location of the receiver will pass through the receiver's filter. Hence, the I-factor is defined as 1 for two channels i and j if $i = j$.

For overlapping channels, an intuitive definition of the I-factor is given in [24, 25] as follows: If P_i is the received power of a particular signal (sent by a transmitter on channel j) at a particular location by a receiver tuned to channel i, and P_j is the received power of the same signal at the same location by a receiver tuned to channel j, then the I-factor of channel j on channel i is defined as:

$$I(i, j) = \frac{P_i}{P_j}. \tag{4.1}$$

Two important properties of the I-factor definition given above are: a) it is not a *commutative* operation, i.e., $I(i, j)$ is not necessarily equal to $I(j, i)$, and b) $I(i, j) \in [0, 1]$ (assuming the transmitter is transmitting on channel j).

It may also be possible to define the I-factor analytically. One such approach for the interference factor of 802.11 DSSS channels appears in [25]. According to this model, the I-factor for two (overlapping) 802.11 channels is calculated as a function of the power spectral distribution of the DSSS signal and the receiver filter's frequency response, as follows:

$$\begin{aligned} I_{theory}(i,j) &= IF_{(T,R)}(5\,|i-j|) \\ &= \int_{-\infty}^{\infty} P_{DSSS}(f, F_{c,t})\; B_R(f, F_{c,t} - 5\,|i-j|)\; df, \end{aligned} \tag{4.2}$$

where $P_{DSSS}(f, F_{c,t})$ is the power spectral distribution function for the interferer DSSS signal, with a center frequency of $F_{c,t}$ (in MHz), and $B_R(f, F_{c,r})$ is the receiver bandpass filter's frequency response, with $F_{c,r} = F_{c,t} - 5\,|i-j|$ (in MHz). In this idealized discrete model, the transmitted signal's power distribution ($P_{DSSS}(f, F_{c,t})$) and the receiver filter ($B_R(f, F_{c,r})$) are approximated with the DSSS TSM defined by the standard, with center frequencies of $F_{c,t}$ and $F_{c,r}$, respectively [25].

The first I-factor definition we propose is the ratio of the area below the intersection of the interferer and the receiver channel signal traces on a spectrum analyzer to the total area below the interferer's signal trace, and is expressed mathematically as follows for interferer channel i and receiver channel j:

$$I(i,j) = \frac{\int_{f_l}^{f_u} \min\left\{P_{intf}(f, F_{c,i}), P_{recv\_ch}(f, F_{c,j})\right\} df}{\int_{f_l}^{f_u} P_{intf}(f, F_{c,i}) df}, \tag{4.3}$$

where $P_{intf}(f, F_{c,i})$ is the interferer signal's power spectral distribution trace captured with a spectrum analyzer, and $P_{recv\_ch}(f, F_{c,j})$ is the power spectral distribution trace captured with a spectrum analyzer that belongs to a transmitted signal using the same standard as the receiver radio and that is on the same channel as the receiver (channel j). $F_{c,i}$ is the center frequency of the interferer signal's channel, which is specific to the standard the interferer radio belongs to. Similarly, $F_{c,j}$ is the center frequency of the receiver's channel, which is specific to the standard the receiver radio belongs to. The interferer and the receiver radios might belong to different wireless communication standards as well as to the same standard. The unit of power used in the calculation of $I(i,j)$ in (4.3) is the unit used in the spectrum analyzer's traces. The lower and upper limits of the integrations (f_l and f_u, respectively) are determined by the total spectrum band covered by the traces of the interferer and the receiver channel signals. We call this method the Signal Intersection Area Method (SIAM) and detail it in Section 4.3.

The second I-factor definition we propose for quantifying interference between the interferer's channel i and the receiver's channel j is the ratio of the total received interference energy on channel j radiated from a transmitter on channel i to the total energy received from the same transmitter by a receiver on channel i. Equation (4.4) expresses this ratio in mathematical terms:

$$I(i,j) = \frac{\int_{f_l}^{f_u} P_{intf}(f, F_{c,i})\ B_R(f, F_{c,j})\ df}{\int_{f_l+(F_{c,i}-F_{c,j})}^{f_u+(F_{c,i}-F_{c,j})} P_{intf}(f, F_{c,i})\ B_R(f, F_{c,i})\ df}, \tag{4.4}$$

where the definitions of $P_{intf}(f, F_{c,i})$, $F_{c,i}$, and $F_{c,j}$ are as in (4.3). B_R is the receiver bandpass filter's frequency response. In (4.4), we use this response twice, once for a receiver tuned to the channel with the center frequency, $F_{c,j}$, and once for another receiver tuned to the same channel as the interferer itself, $F_{c,i}$. We obtain $P_{intf}(f, F_{c,i})$ experimentally using a spectrum analyzer. The lower limit of integration, f_l, for the receiver channel j is $f_l = F_{c,j} - \frac{Bw}{2}$, and the upper limit of integration, f_u, for the receiver channel j is $f_u = F_{c,j} + \frac{Bw}{2}$, where Bw is the width of the bandpass filter's response. This method, which we call Percentage of Maximum Interference Energy (PMIE), has a more concrete physical interpretation that we discuss in detail in Section 4.3.

4.3 Our Proposed Interference Factor Calculation Methods

The method we follow to compute the I-factor is based on physical-layer measurements in the frequency domain, taken with a spectrum analyzer. Using the spectrum analyzer, we obtain signal traces showing the power spectral distribution of a transmitted wireless signal. To capture channel activity, we generate and analyze 802.11b/g packets using the multi-radio mesh nodes in our testbed BilMesh, which is a multi-hop, multi-radio wireless mesh networking testbed we have established at Bilkent University. To analyze the 802.15.4 interference on 802.11 receivers, we add ZigBee motes to our testbed. We use a Yellowjacket-Tablet 2.4/5 GHz Wi-Fi spectrum analyzer [69] to obtain the signal traces. The

setup required for the measurements consists of a single wireless transmitter. However, as explained in Section 4.4.3, two or more transmitters operating on the same channel can be used to shorten the time needed to collect the signal traces. The transmitter(s') channel is set in succession to one of the channels of interest between which the I-factor is to be calculated.

After collecting the traces for the receiver and interferer channel signals, we consider the area below the intersection of these traces in relation to the power spectral density of the interferer signal. The ratio is considered to be the I-factor between two channels. This I-factor definition has been introduced formally in Section 4.2 as SIAM. Below, we describe the steps showing how it can be computed:

1. We obtain the spectrum analyzer traces for the two wireless signals between which the I-factor is to be calculated. Because I-factor is defined as the ratio of the received energy on a receiver channel to the radiated energy on a transmitter (interferer) channel, we have to obtain the interferer's power spectral distribution using the spectrum analyzer. In our current implementation, we obtain, store, and process this information with device independent bitmap files. We use bitmap files because the spectrum analyzer available to us can export this data in bitmap format; however any other form of representation for the power spectral density data can be used with our method.

2. Once the two signal traces are obtained, the area of their intersection in the frequency domain over a reference power level is calculated. In our current implementation, the reference power level is determined by the user, considering the noise power level available in the spectrum analyzer data. The user also determines the interval in the frequency domain over which the integration will be performed.

3. The area in the frequency domain below the interferer signal and above the reference power level used in the previous step is calculated. The same frequency interval as in the previous step is used for the integration.

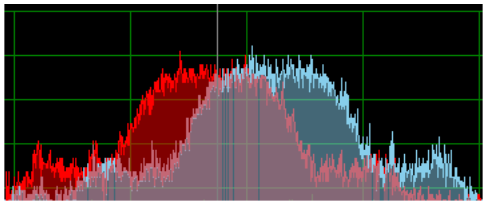
4. The I-factor is obtained by dividing the area below the intersection of the two signals by the area below the interferer signal.

In the second step of the above procedure, we assume that the receiver radio uses a filter similar to its TSM and we estimate this filter using its transmit power distribution over the frequency domain. With this estimation, we calculate the interferer power falling below the measured (estimated) filter, as in Figure 4.1(a).

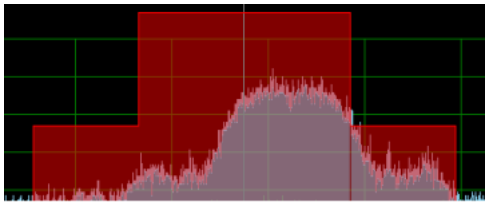
If we have a better estimation for the filter being used by the receiver radio, there is no need to collect the traces for the receiver in the first step. Instead, after collecting the interferer traces, we can directly calculate the ratio of the interferer signal power falling below this filter to the total interferer power to estimate the I-factor, as in Figure 4.1(b).

Algorithm 1 outlines the steps of our proposed method. The output of the algorithm, $I(ch_{intf}, ch_{recv})$, is the I-factor calculated between channels ch_{intf} and ch_{recv}. If there is an analytical model for the receiver bandpass filter's frequency response, such as the one in (4.5), it must be converted into a suitable representation for processing. In our implementation, this corresponds to adding the filter model onto the bitmap where the interferer signal trace resides, using a different color than the interferer signal trace's color. To calculate the total interferer power and the amount of interferer power that overlaps with the receiver radio's filter on the frequency interval $[f_l, f_u]$, we use the procedure given in Algorithm 2. The inputs $[f_l, f_u]$ and *referencePowerLevel* should be specified in terms of the pixels of the bitmap B.

The method we propose above defines I-factor as the ratio of the area below the intersection of two frequency domain signal traces to the total area below the trace that belongs to the interferer signal. We now discuss another definition for the I-factor that has a more concrete physical interpretation. Equation (4.4), introduced earlier, models the percentage of the maximum interference energy radiated on channel i and received by a receiver on channel j, where $F_{c,i}$ is the center frequency of channel i and $F_{c,j}$ is the center frequency of channel j. The maximum interference energy is defined as the interference energy that would be



(a) Two DSSS signals shown together. The receiver filter may be estimated using a transmission on the receiver's channel (channel 6 here, shown in red). The blue trace belongs to an interferer on the adjacent channel 7.



(b) The receiver filter, in red, estimated as the DSSS TSM centered on channel 6 and the captured interferer DSSS signal, in blue, on channel 7, shown together.

Figure 4.1: I-factor can be modeled when no analytical model is given for the receiver filter's frequency response by estimating the filter by the TSM as in 4.1(a). When an analytical model is assumed for the receiver filter's frequency response, the I-factor can be modeled without the need for receiver channel traces, as in 4.1(b).

Algorithm 1 SIAM I-Factor Model and Computation Method
Input: $F_{c,i}$. The center frequency of the interferer channel, i.
Input: $F_{c,j}$. The center frequency of the receiver channel, j.
Output: $I(i,j)$

1: Collect spectrum analyzer traces for the interferer signal on channel i
2: **if** No model is assumed for the receiver filter's frequency response **then**
3: Collect spectrum analyzer traces on the receiver channel j
4: **else**
5: Convert the receiver filter model into suitable representation
6: **end if**
7: $[f_l, f_u] \Leftarrow$ User input ▷ f_l and f_u are the limits of integration
8: $totalInterfererPower \Leftarrow \int_{f_l}^{f_u} P_{intf}(f, F_{c,i})df$ ▷ Approximated with Algorithm 2
9: $overlapPower \Leftarrow \int_{f_l}^{f_u} \min\{P_{intf}(f, F_{c,i}), P_{recv\_ch}(f, F_{c,j})\}\, df$ ▷ Approximated with Algorithm 2
10: $I(i,j) \Leftarrow \frac{overlapPower}{totalInterfererPower}$

Algorithm 2 Total Interferer Power and Overlap Power Calculation on Bitmap
Input: Bitmap B containing the trace of the receiver channel signal (or the receiver filter's frequency response curve) together with the trace of the interferer signal. The traces should be in different colors (i.e., $Color_{recv}$ and $Color_{intf}$).
Input: $Color_{recv}$, $Color_{intf}$. The colors of the receiver channel signal trace (or the frequency response curve) and the interferer signal trace, respectively.
Input: *referencePowerLevel*. Power above *referencePowerLevel* is summed.
Input: $[f_l, f_u]$. The frequency bounds of the interval of integration.
Output: *totalInterfererPower*, *overlapPower*

1: $totalInterfererPower \Leftarrow 0$
2: $overlapPower \Leftarrow 0$
3: Draw line $y = referencePowerLevel$ on bitmap B with color $Color_{ref}$, such that $Color_{ref} \notin \{Color_{recv}, Color_{intf}\}$
4: **for** $f = f_l$ `to` f_u **do**
5: $Power_{recv} \Leftarrow 0$
6: $Power_{intf} \Leftarrow 0$
7: **for** $p = 1$ `to` $height[B]$ **do** ▷ $height[B]$ is the height of the bitmap
8: **if** $color[P_{f,p}] = Color_{recv}$ **then** ▷ $color[P_{f,p}]$ is the color of pixel $P_{f,p}$
9: $Power_{recv} \Leftarrow (referencePowerLevel - p)$
10: **else if** $color[P_{f,p}] = Color_{intf}$ **then**
11: $Power_{intf} \Leftarrow (referencePowerLevel - p)$
12: **else if** $color[P_{f,p}] = Color_{ref}$ **then**
13: **break**
14: **end if**
15: **end for**
16: **if** $Power_{intf} \neq 0$ **then**
17: $totalInterfererPower \Leftarrow totalInterfererPower + Power_{intf}$
18: **if** $Power_{recv} \neq 0$ **then** ▷ Overlap region
19: $overlapPower \Leftarrow overlapPower + \min\{Power_{recv}, Power_{intf}\}$
20: **end if**
21: **end if**
22: **end for**

received by another receiver operating on channel i in the same location as the receiver on channel j.

Algorithm 3 outlines the steps of I-factor calculation based on (4.4). We call this method described with Algorithm 3 as the PMIE method: *P*ercentage of *M*aximum *I*nterference *E*nergy. While calculating *totalReceivedIntfEn* and *maximumIntfEn*, Algorithm 4 is called twice, first with a receiver channel signal trace (or the receiver filter's frequency response centered around $F_{c,j}$) and then with an interferer channel signal trace (or the receiver filter's frequency response centered around $F_{c,i}$).

Algorithm 3 PMIE I-Factor Model and Computation Method
Input: $F_{c,i}$. The center frequency of the interferer channel, i.
Input: $F_{c,j}$. The center frequency of the receiver channel, j.
Output: $I(i,j)$

1: Collect spectrum analyzer traces for the interferer signal on channel i
2: **if** No model is assumed for the receiver filter's frequency response **then**
3: Collect spectrum analyzer traces on the receiver channel j
4: $[f_l, f_u] \Leftarrow$ User input ▷ f_l and f_u are the limits of integration
5: **else**
6: Convert the receiver filter model into suitable representation
7: $f_l \Leftarrow F_{c,j} - \frac{Bw}{2}$ ▷ Bw is the width of the bandpass filter's frequency response
8: $f_u \Leftarrow F_{c,j} + \frac{Bw}{2}$ ▷ f_l and f_u are the limits of integration
9: **end if**
10: $totalReceivedIntfEn \Leftarrow \int_{f_l}^{f_u} P_{intf}(f, F_{c,i})\, B_R(f, F_{c,j})\, df$ ▷ Approximated with Algorithm 4
11: $maximumIntfEn \Leftarrow \int_{f_l+(F_{c,i}-F_{c,j})}^{f_u+(F_{c,i}-F_{c,j})} P_{intf}(f, F_{c,i})\, B_R(f, F_{c,i})\, df$ ▷ Approximated with Algorithm 4
12: $I(i,j) \Leftarrow \frac{totalReceivedIntfEn}{maximumIntfEn}$

We implement these novel methods for modeling and computing the I-factor using the Java programming language. The current implementation requires the signal traces on the receiver's channel and on the interferer's channels to be placed on the same bitmap using different colors, as in Figures 4.3 through 4.7. Using the graphical user interface (GUI) of the Java program, the user is able to determine the reference power level and the integration interval in the frequency domain. The program outputs the ratio of the intersection area to the total power of the first signal and the ratio of the intersection area to the total power of the second signal. I-factor values with respect to a receiver operating on the first signal's channel are the ratios of the intersection areas to the second signal's total power. Likewise, I-factor values with respect to a receiver operating on the second signal's

Algorithm 4 Received Energy Calculation on Bitmap
Input: Bitmap B containing the trace of the receiver channel signal (or the receiver filter's frequency response curve) together with the trace of the interferer signal. The traces should be in different colors (i.e., $Color_{recv}$ and $Color_{intf}$).
Input: $Color_{recv}$, $Color_{intf}$. The colors of the receiver channel signal trace (or the frequency response curve) and the interferer signal trace, respectively.
Input: $[f_l, f_u]$. The frequency bounds of the interval of integration.
Output: *totalReceivedEn*

```
1:  totalReceivedEn ⇐ 0
2:  for f = f_l to f_u do
3:     Power_recv ⇐ 0
4:     Power_intf ⇐ 0
5:     for p = 1 to height[B] do                ▷ height[B] is the height of the bitmap
6:        if color[P_f,p] = Color_recv then     ▷ color[P_f,p] is the color of pixel P_f,p
7:           Power_recv ⇐ 10 log(p/10)
8:        else if color[P_f,p] = Color_intf then
9:           Power_intf ⇐ 10 log(p/10)
10:       end if
11:    end for
12:    totalReceivedEn ⇐ Power_recv Power_intf
13: end for
```

channel are the ratios of the intersection areas to the first signal's total power.

PMIE has a more concrete physical interpretation when compared to SIAM because PMIE defines the I-factor as the ratio of the total received interference energy from an interferer on the receiver's channel to the total received energy from the same interferer on the interferer's channel (the maximum interference energy). PMIE I-factor value is, consequently, strictly 1 when the receiver captures all of the interferer's energy and 0 when the receiver filters all of the interferer's energy. However, because of the way SIAM is defined, these may not always hold for SIAM I-factor values. PMIE, on the other hand, requires the knowledge of the receiver radio filter's frequency response (the B_R function). If B_R is not known, SIAM, however, can approximate it from transmissions on the receiver radio's channel by a transmitter radio using the same wireless standard and PHY layer specification as the receiver radio. This gives more flexibility to SIAM and makes it possible to obtain approximated I-factor values using SIAM when B_R is not known. If B_R is known, PMIE can be used to obtain more accurate I-factor values.

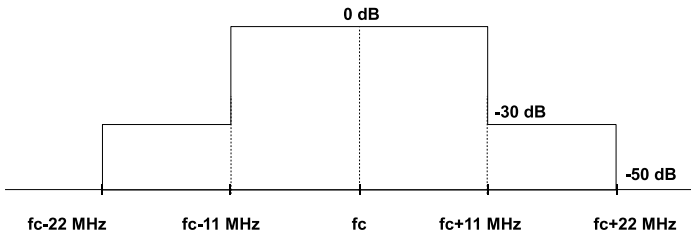


Figure 4.2: IEEE 802.11 DSSS Transmit Spectrum Mask.

Our measurement results for the interference factor between two DSSS signals and between the DSSS TSM and an interferer DSSS signal are given in Section 4.4.1. We have written a separate Java program to implement PMIE (Algorithms 3 and 4). In Section 4.4.2, we report on the I-factor calculations using the PMIE method. In Section 4.4.3, we report our results on the I-factor values between an 802.15.4 OQPSK interferer and an 802.11 DSSS receiver.

4.4 Measurement Results and Comparisons

4.4.1 Measurements for Modeling Interference Between 802.11 DSSS Signals using SIAM

We collect 802.11b DSSS signal traces (see Figures 4.3-4.8) with the 2.4 GHz spectrum analyzer. For collecting these traces, one radio is kept fixed at channel 6 (2437 MHz) and the other radio is swept from channel 7 (2442 MHz) to channel 12 (2467 MHz) of the 14 channels defined by the standard. In this case, we assume that the receiver filter for the specific DSSS radio in use is neither known, nor can it be estimated analytically. We also consider the case where we approximate the receiver filter using the DSSS TSM, depicted in Figure 4.2 and defined as follows:

Ch. Separation	Signal Traces (SIAM)	TSM Assumption (SIAM)	Signal Traces (PMIE)	Analytical Model	SNR Measurements
0	-	-	1	-	-
1	0.61	0.82	0.68	0.60	0.96
2	0.33	0.56	0.18	0.30	0.77
3	0.14	0.27	0.008	0.11	0.66
4	0.09	0.1	0.003	0.01	0.39
5	0.03	0.03	0.002	0	0
6	0.01	0.01	0.002	-	-

Table 4.1: Interference factors calculated using SIAM and PMIE (see Figures 4.3-4.14 and 4.16-4.22) and compared with some of the existing models in the literature.

$$TSM(f, F_c) = \begin{cases} -50\,dBr & \text{if } |f - F_c| > 22\ MHz \\ -30\,dBr & \text{if } 11 < |f - F_c| < 22\ MHz \\ 0\,dBr & \text{Otherwise.} \end{cases} \tag{4.5}$$

Here, f denotes the frequency and F_c is the center frequency of the receiver channel. With this approximation, there is no need to collect traces for a transmitted signal on the receiver's channel; it suffices to collect traces for the interferer radio. Figures 4.9 to 4.14 show the data for these measurements. In these measurements, the receiver is kept fixed at channel 6 (center frequency at 2437 MHz) and the interferer sweeps from channel 7 to channel 12. Only the interferer signal traces need be collected; the receiver filter's frequency response curve is approximated by the 802.11 DSSS TSM centered at channel 6, shown in red color in these figures.

The I-factor values calculated for the above two cases using SIAM are given in the second and third columns of Table 4.1. In the fifth column, we give the estimations of the I-factor for DSSS radios using the analytical model proposed in [25]. In this model, the receiver filter's frequency response ($B_R(f, F_{c,r})$ in (4.2)) is assumed to be identical to the DSSS TSM and is given in (4.5) ($B_R(f, F_{c,r}) = TSM(f, F_{c,r})$). The last column of Table 4.1 gives the SNR measurement-based estimation of the I-factor values as reported in [24], measured for the cases where the receiver is fixed on channel 6 and the transmitter is operated on channels 7 to 11. These transmitter channels match the channels used in our experiments.

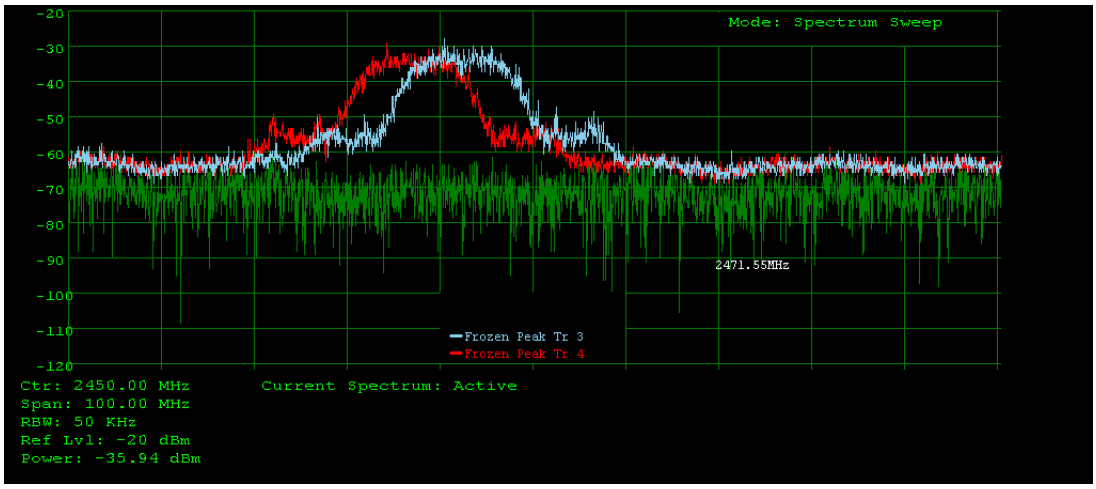


Figure 4.3: Signal traces showing overlap between transmitted signals on channels 6 (red trace) and 7 (blue trace).

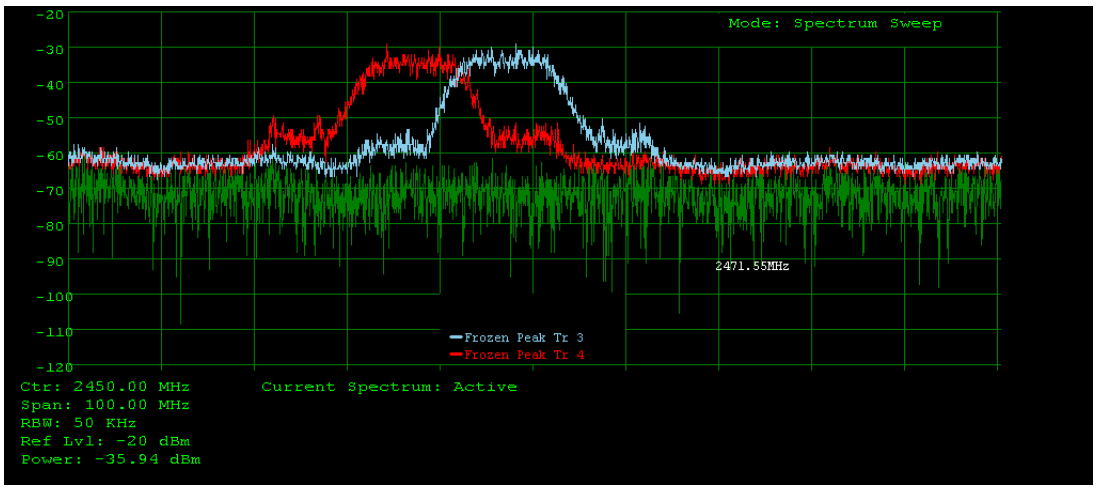


Figure 4.4: Signal traces showing overlap between transmitted signals on channels 6 (red trace) and 8 (blue trace).

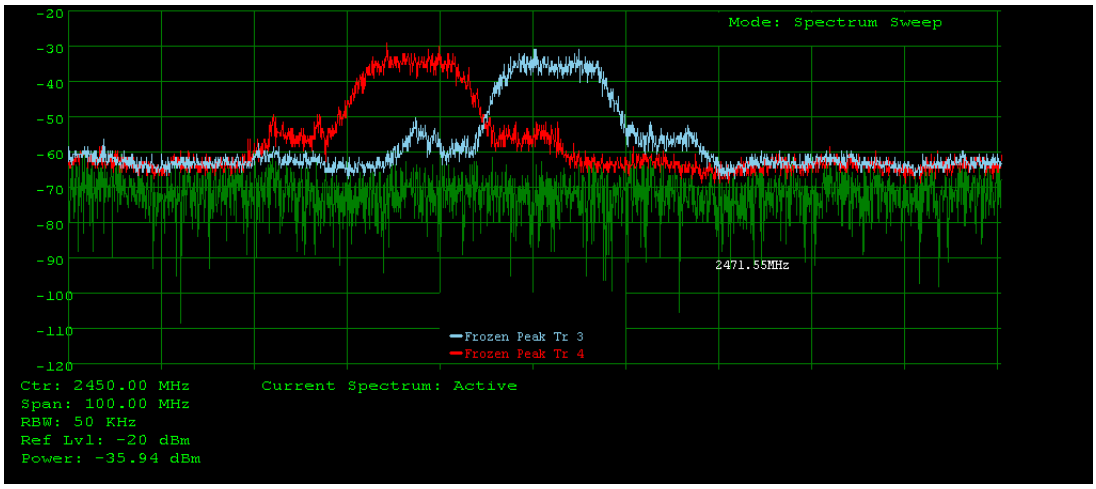


Figure 4.5: Signal traces showing overlap between transmitted signals on channels 6 (red trace) and 9 (blue trace).

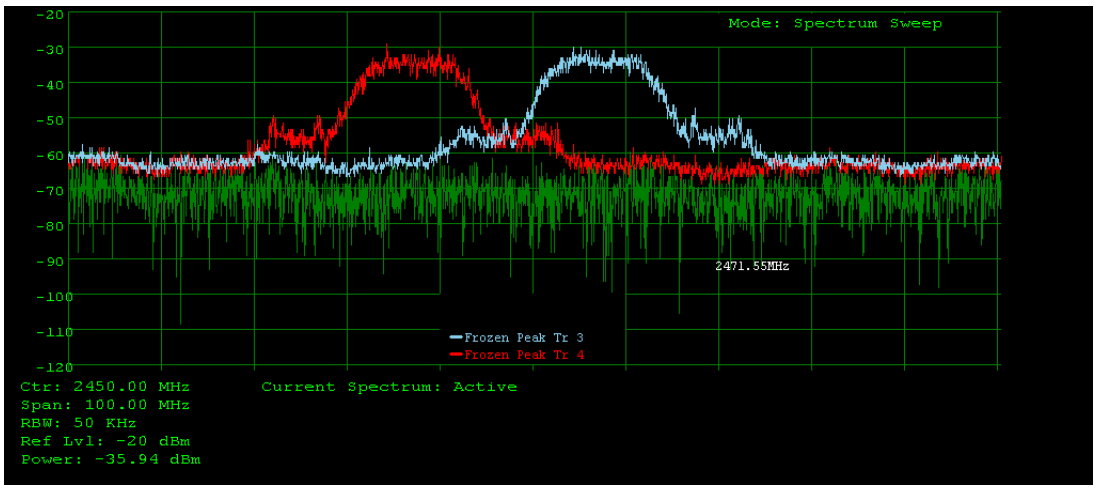


Figure 4.6: Signal traces showing overlap between transmitted signals on channels 6 (red trace) and 10 (blue trace).

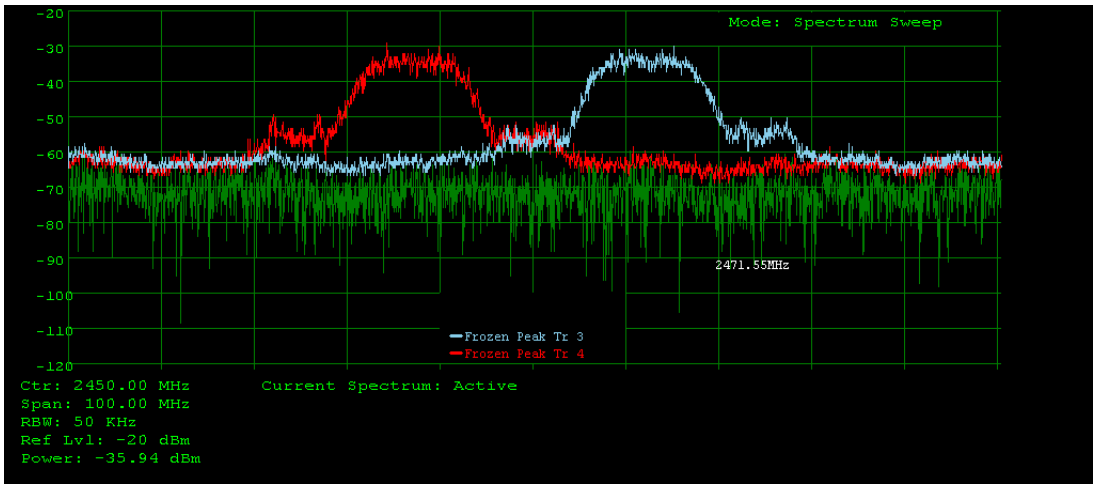


Figure 4.7: Signal traces showing overlap between transmitted signals on channels 6 (red trace) and 11 (blue trace).

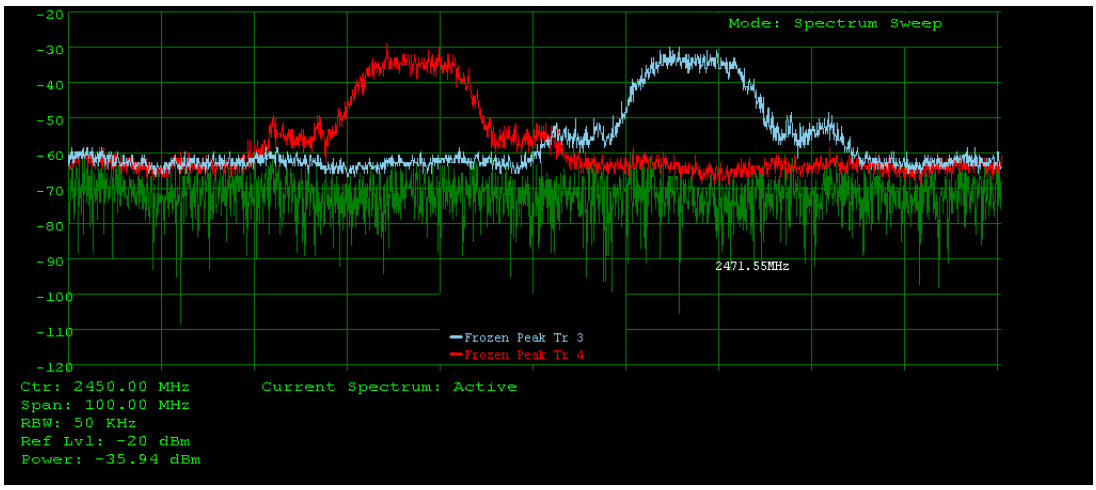


Figure 4.8: Signal traces showing overlap between transmitted signals on channels 6 (red trace) and 12 (blue trace).

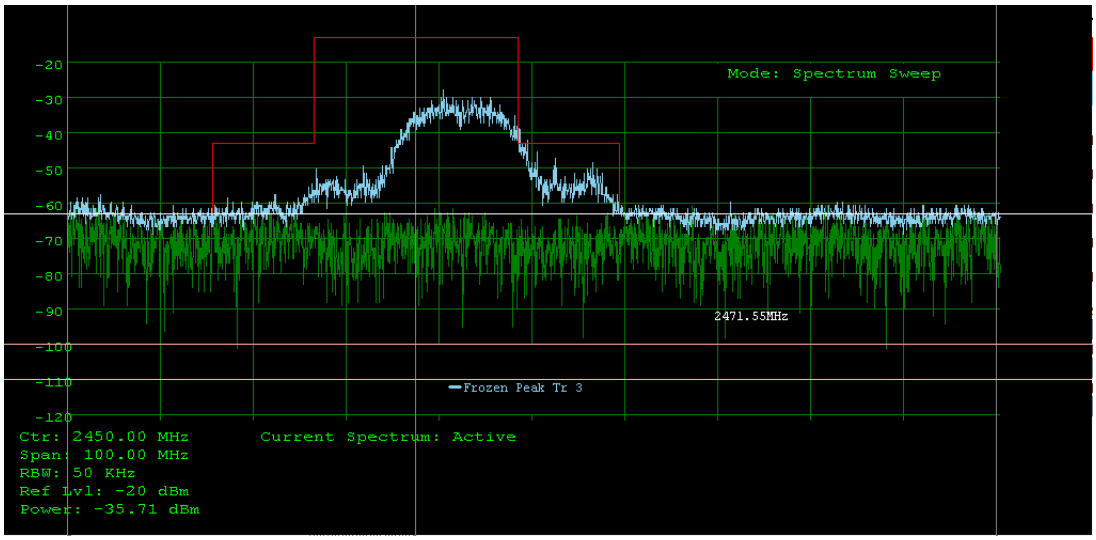


Figure 4.9: Transmit spectrum mask on channel 6 (red) and signal trace on channel 7 (blue).

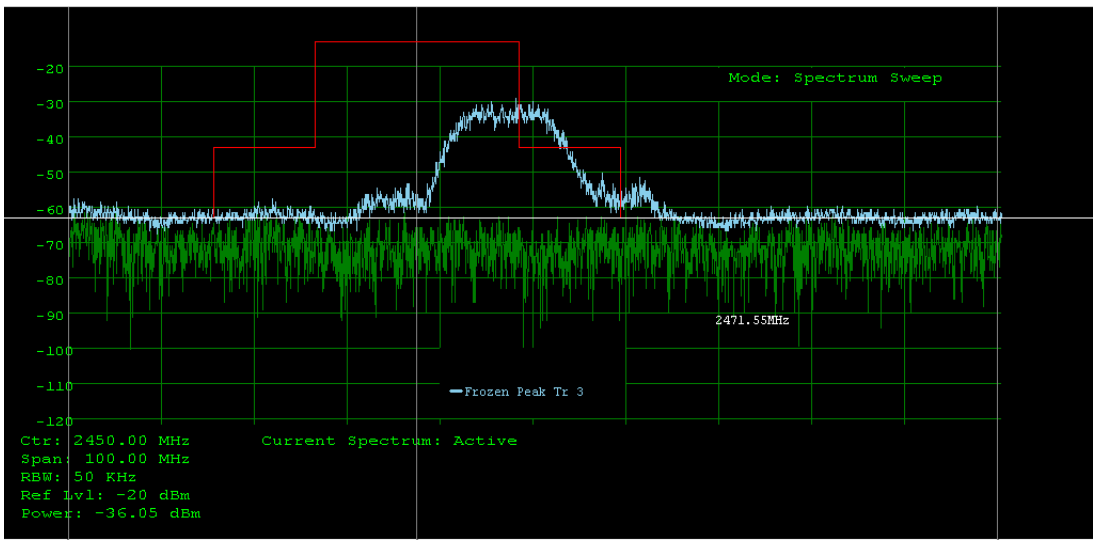


Figure 4.10: Transmit spectrum mask on channel 6 (red) and signal trace on channel 8 (blue).

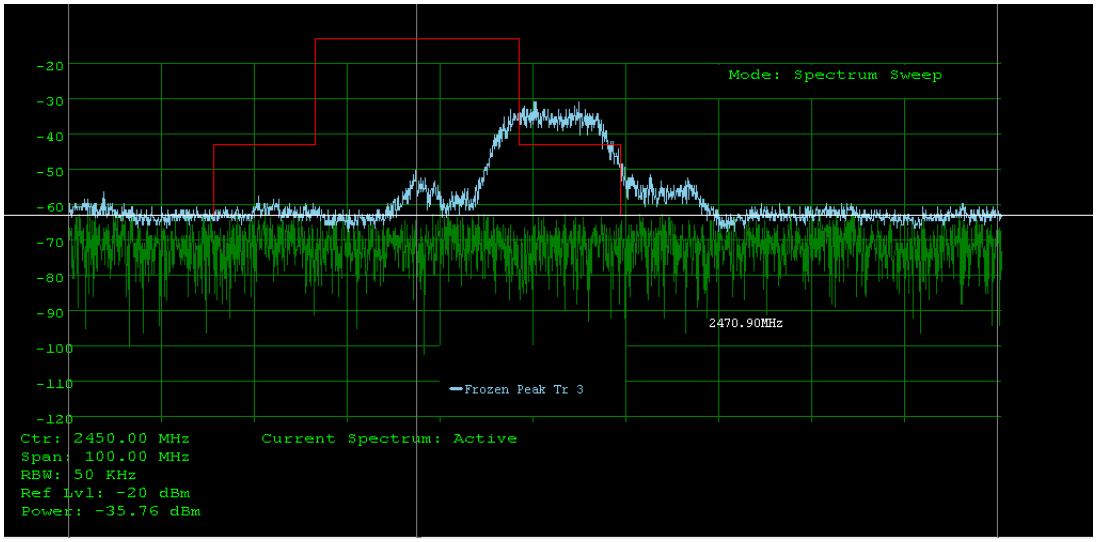


Figure 4.11: Transmit spectrum mask on channel 6 (red) and signal trace on channel 9 (blue).

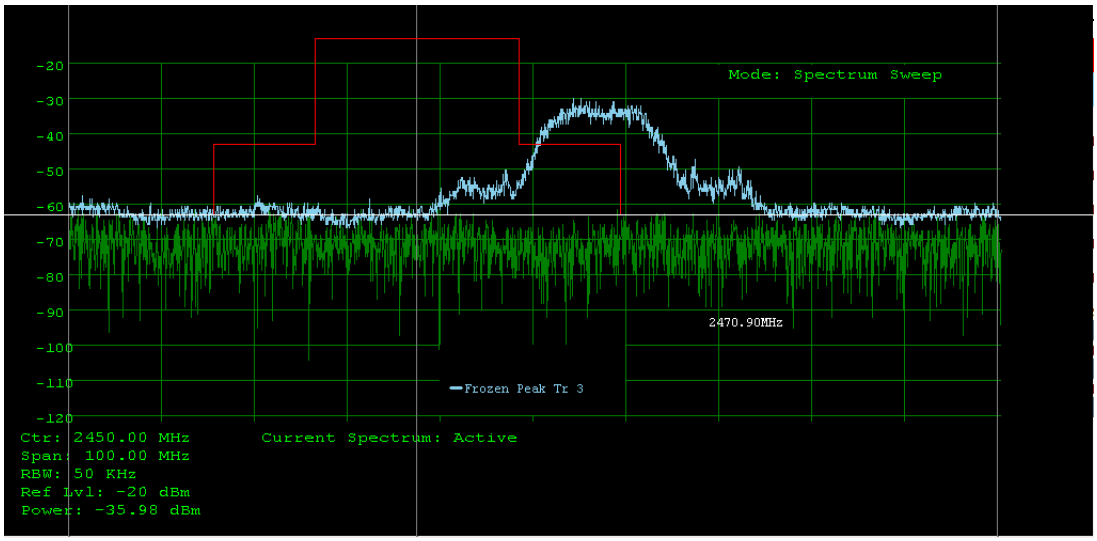


Figure 4.12: Transmit spectrum mask on channel 6 (red) and signal trace on channel 10 (blue).

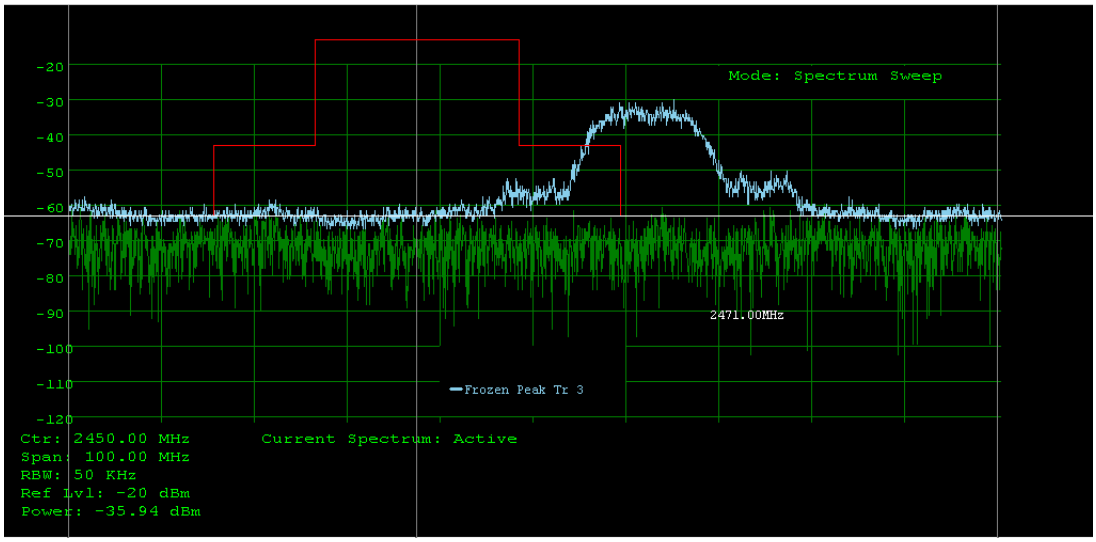


Figure 4.13: Transmit spectrum mask on channel 6 (red) and signal trace on channel 11 (blue).

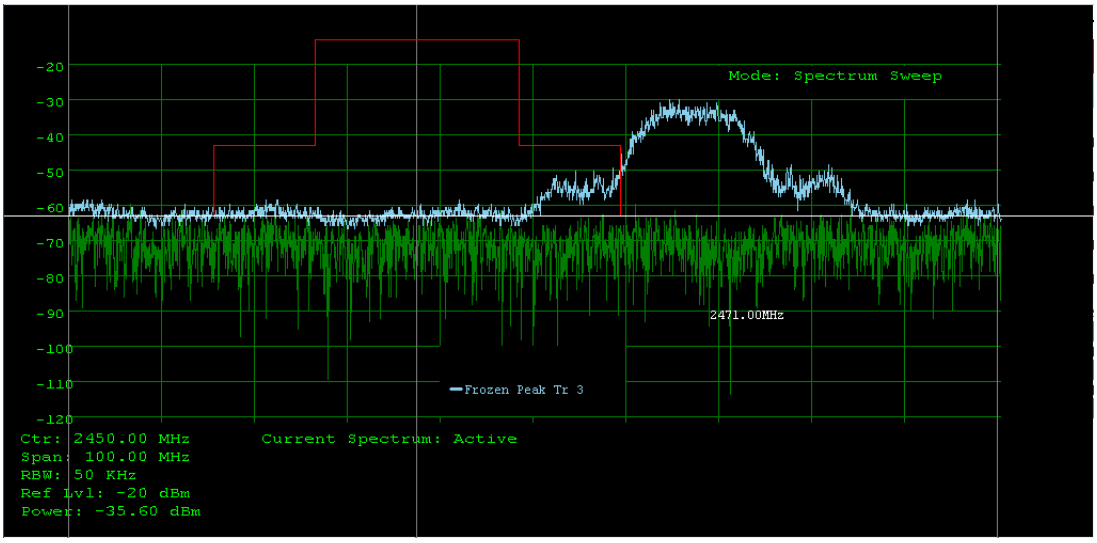


Figure 4.14: Transmit spectrum mask on channel 6 (red) and signal trace on channel 12 (blue).

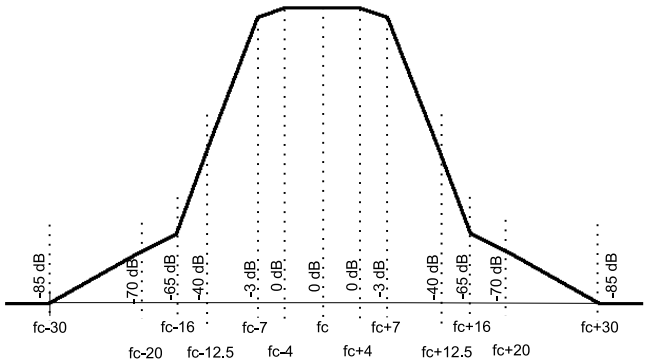


Figure 4.15: Receiver filter's frequency response of the Maxim MAX2820/MAX2821 802.11b transceiver. f_c denotes the center frequency, and the unit of the x-axis is MHz.

4.4.2 Measurements for Modeling Interference Between 802.11 DSSS Signals using the PMIE Method

Using the collected interferer 802.11 DSSS signal traces and the receiver filter's frequency response of the Maxim MAX2820/2821 802.11b transceiver [70] given in Figure 4.15, we calculate the I-factors between 802.11 DSSS channels using Algorithms 3 and 4 (PMIE). Figures 4.16 to 4.22 show the data used for these calculations. For the related measurements, the receiver is kept fixed at channel 6 (represented by the frequency response curve in red centered at 2437 MHz) and the interferer sweeps from channel 6 to channel 12. For this setup, only the interferer signal traces need be collected; the receiver filter's frequency response curve is generated by our software. The fourth column of Table 4.1 summarizes the I-factor calculations for this setup. Compared to the third column, which uses the 802.11 DSSS TSM and SIAM, it can be observed that the I-factor values obtained using the narrower MAX2820/2821 filter frequency response and the PMIE method drop faster as channel separation increases. Further, by the definition of I-factor in (4.4), we have an I-factor exactly equal to 1.0 when the channel separation is 0.

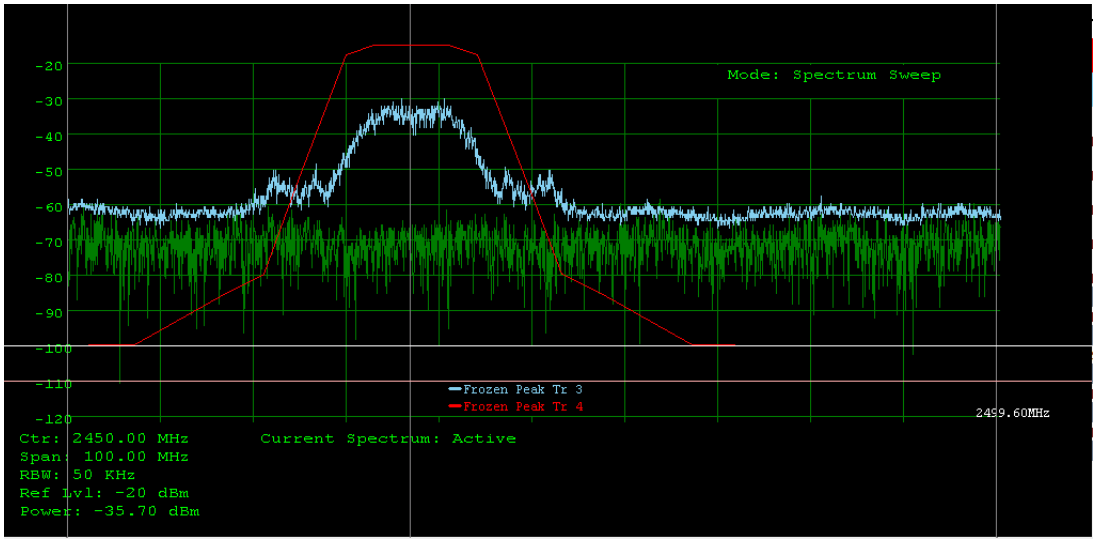


Figure 4.16: MAX2820 receiver filter's frequency response on channel 6 (red) and signal trace on channel 6 (blue).

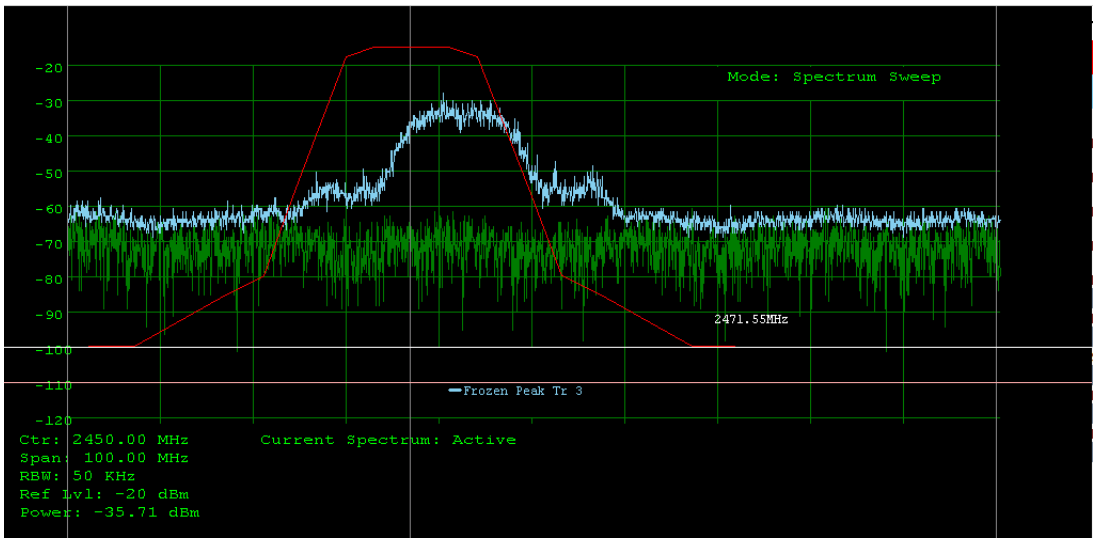


Figure 4.17: MAX2820 receiver filter's frequency response on channel 6 (red) and signal trace on channel 7 (blue).

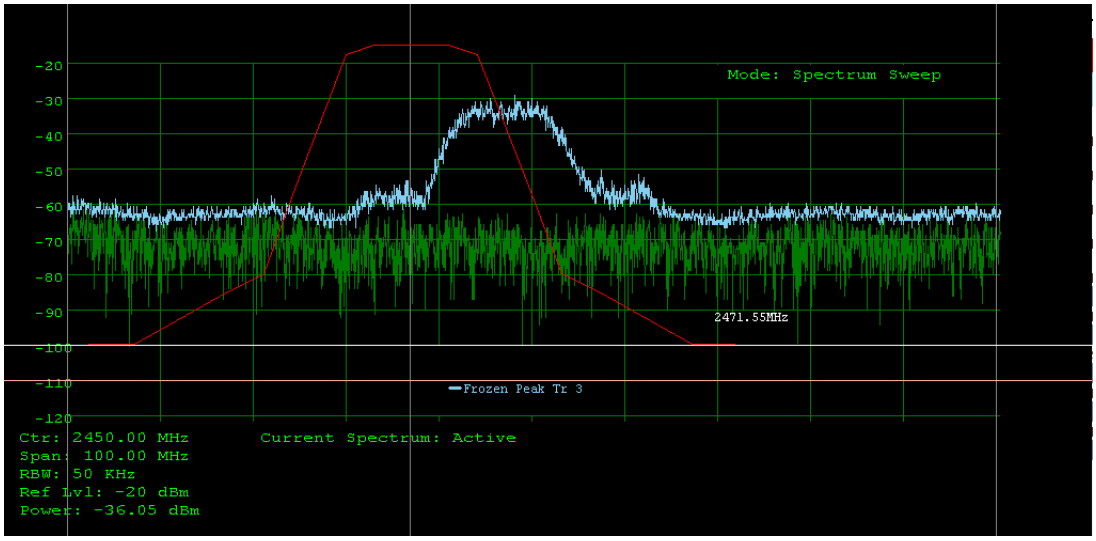


Figure 4.18: MAX2820 receiver filter's frequency response on channel 6 (red) and signal trace on channel 8 (blue).

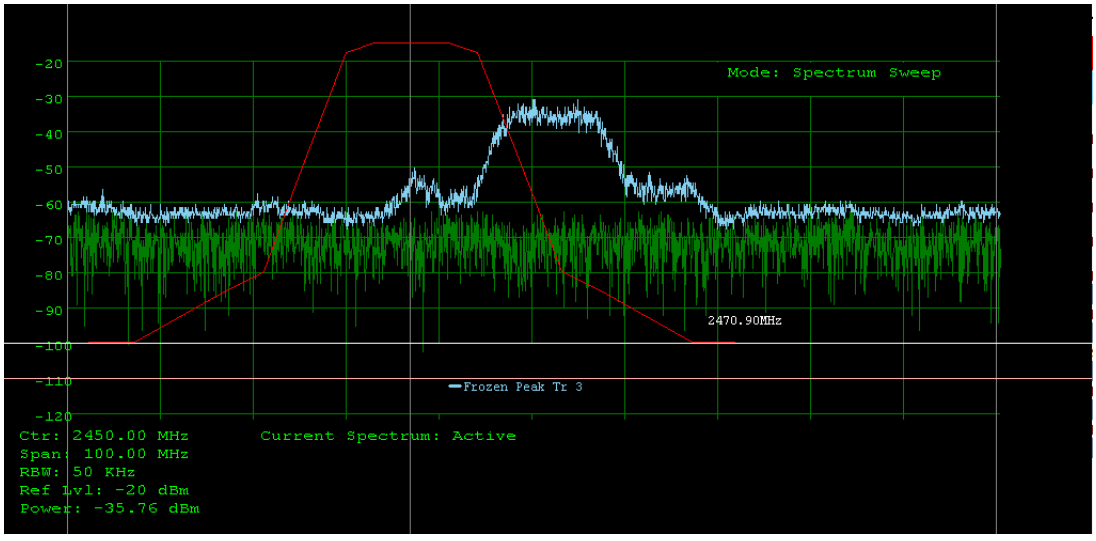


Figure 4.19: MAX2820 receiver filter's frequency response on channel 6 (red) and signal trace on channel 9 (blue).

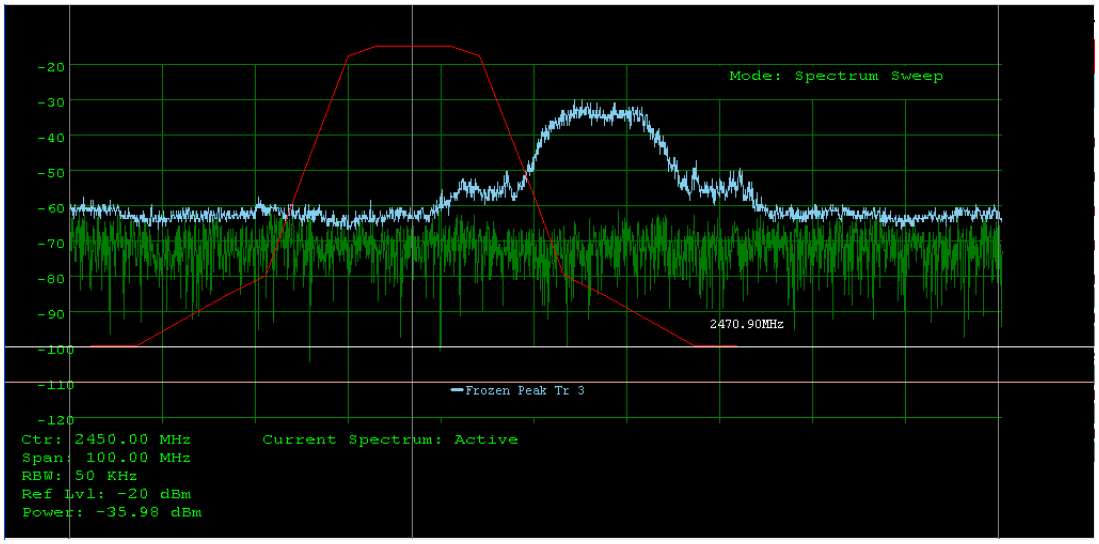


Figure 4.20: MAX2820 receiver filter's frequency response on channel 6 (red) and signal trace on channel 10 (blue).

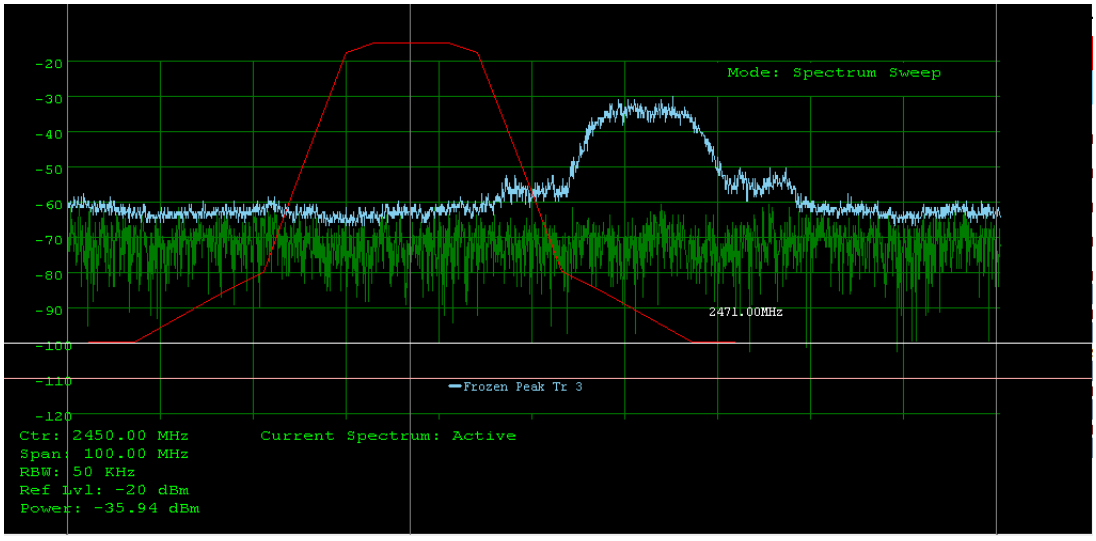


Figure 4.21: MAX2820 receiver filter's frequency response on channel 6 (red) and signal trace on channel 11 (blue).

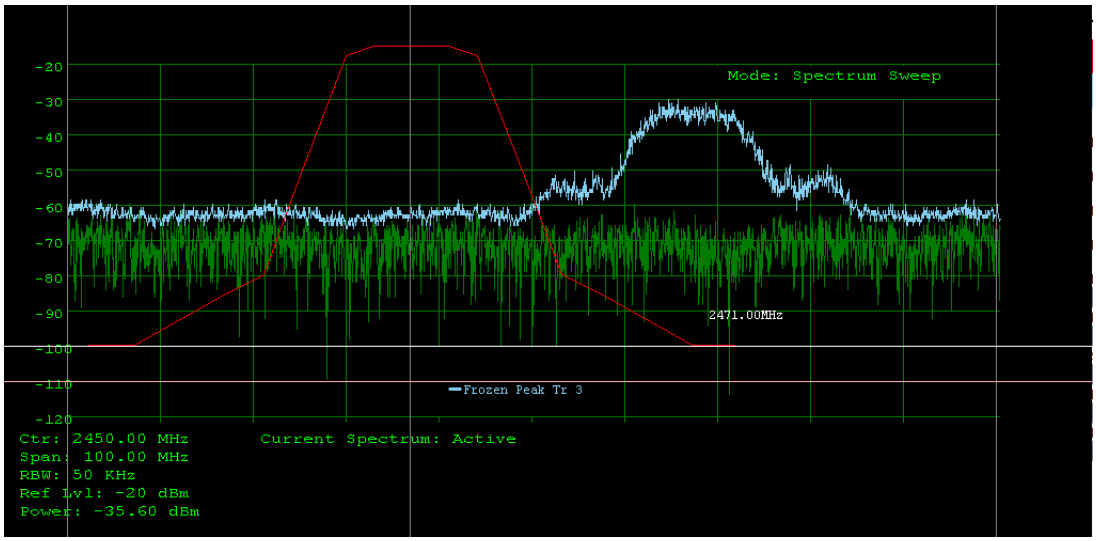


Figure 4.22: MAX2820 receiver filter's frequency response on channel 6 (red) and signal trace on channel 12 (blue).

4.4.3 Measurements for Modeling Interference Between 802.11 DSSS and 802.15.4 OQPSK Signals

To model the interference between an 802.15.4 (ZigBee) transmitter and an 802.11b receiver, we perform another set of experiments and take measurements with our spectrum analyzer. In this set, we use five Crossbow TelosB motes [71] and two 802.11b radios in ad-hoc (IBSS) mode. The TelosB motes have IEEE 802.15.4 compliant Texas Instruments CC2420 radios with integrated onboard antennas, and communicate with a data rate of 250 Kbps. They are commonly employed in wireless sensor networks. One of the TelosB motes is used as the *base station* in our experiment. The other four motes periodically sample their sensors and transmit their readings to the base station. We use four ZigBee transmitters to increase the chance of an 802.15.4 signal being registered on the spectrum analyzer. The 802.11b radios are operated on (802.11) channel 3, whose center frequency is 2422 MHz. The channel of the ZigBee network is varied from (802.15.4) channel 11 (central frequency 2405 MHz) to channel 17 (central frequency 2435 MHz).

The IEEE 802.15.4 standard [72] defines channels in the 868/915 MHz and

2450 MHz spectra. The frequency band in the 868 MHz spectrum is narrow, starting at 868 MHz and ending at 868.6 MHz; and the frequency band in the 915 MHz spectrum starts at 902 and ends at 928 MHz. The frequency band that we observe in our experiments is in the 2.4 GHz spectrum, and starts at 2400 MHz, and ends at 2483.5 MHz. The IEEE 802.15.4 2006 standard specifies channels with a combination of a channel *page* and a channel *number*. For channel page 0, the standard defines 16 channels in the 2450 MHz band, 10 channels in the 915 MHz band, and one channel in the 868 MHz band. Each consecutive 802.15.4 channel on channel page 0 in the 2450 MHz band is separated by 5 MHz, and the center frequency, F_c, for channel k is given by [72]:

$$F_c = 2405 + 5(k - 11), k = 11, 12, \ldots, 26, \tag{4.6}$$

where F_c is in MHz. The first channel on channel page 0 in the 2450 MHz band is channel 11 and the last channel is channel 26.

We generate 802.11 traffic by ping flooding. During the experiment, we collect spectrum analyzer traces (see Figures 4.23-4.29, where the receiver is an 802.11 radio operating on channel 3 (shown in blue) and the interferer radio is a ZigBee radio operating on channels 11 through 17 (shown in red)).

Table 4.2 summarizes the I-factor calculations using SIAM for these experiments. The first column shows that the 802.11 receiver is fixed at channel 3. The second column shows the 802.15.4 channel of the ZigBee interferer. The last column shows the I-factor values computed with our method. When the channel center frequencies of the 802.15.4 and 802.11 radios are separated by at least 17 MHz, the interference power that leaks from the 802.15.4 radio on the 802.11 receiver is 0. If the center frequencies are separated by at least 12 MHz, the interference power from the 802.15.4 radio is still negligible. However, if the center frequencies are separated by less than 12 MHz, there is significant power leakage from the 802.15.4 radio.

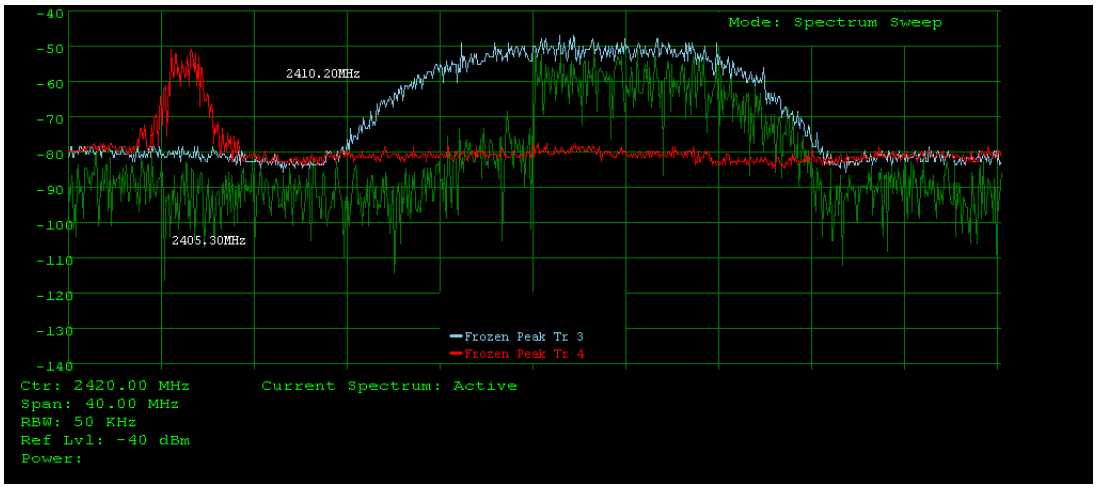


Figure 4.23: Signal traces showing overlap between 802.11 channel 3 (blue) and 802.15.4 channel 11 (red).

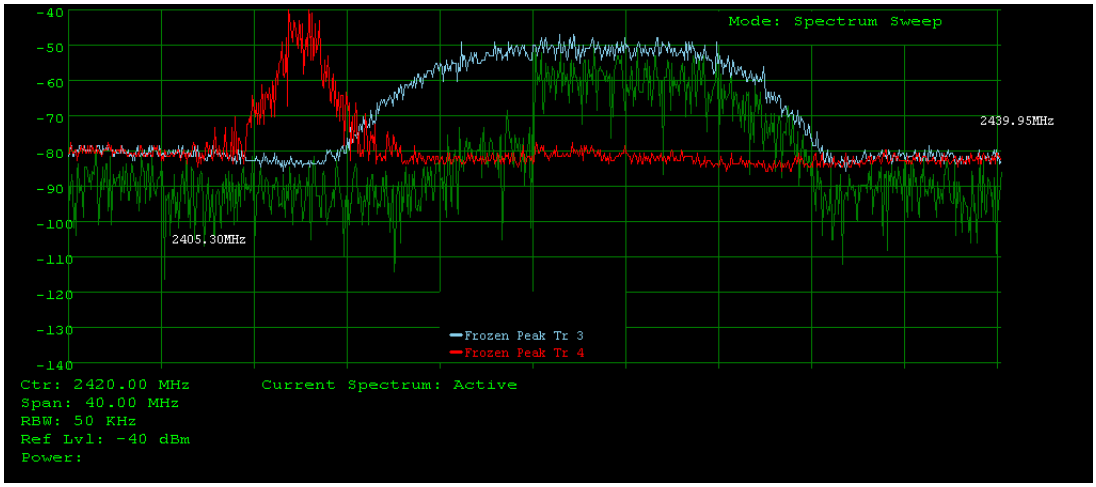


Figure 4.24: Signal traces showing overlap between 802.11 channel 3 (blue) and 802.15.4 channel 12 (red).

802.11 Channel	**ZigBee Channel**	**I-Factor Based on Spectrum Analyzer Traces**
3	11	0
3	12	0.01
3	13	0.84
3	14	0.92
3	15	0.96
3	16	0.55
3	17	0.02

Table 4.2: Interference factors calculated using SIAM (see Figures 4.23-4.29). The ZigBee radio is the interferer to the 802.11 radio on channel 3.

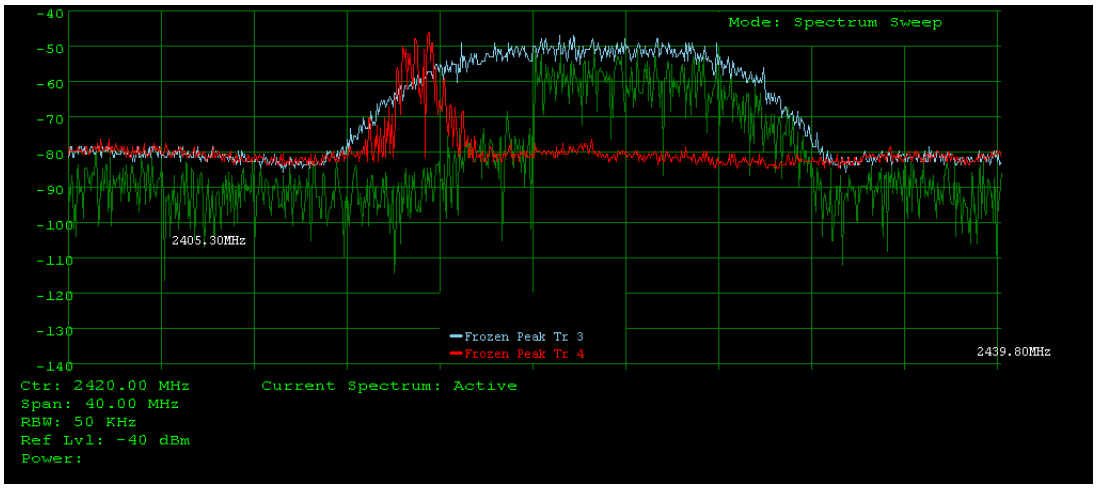


Figure 4.25: Signal traces showing overlap between 802.11 channel 3 (blue) and 802.15.4 channel 13 (red).

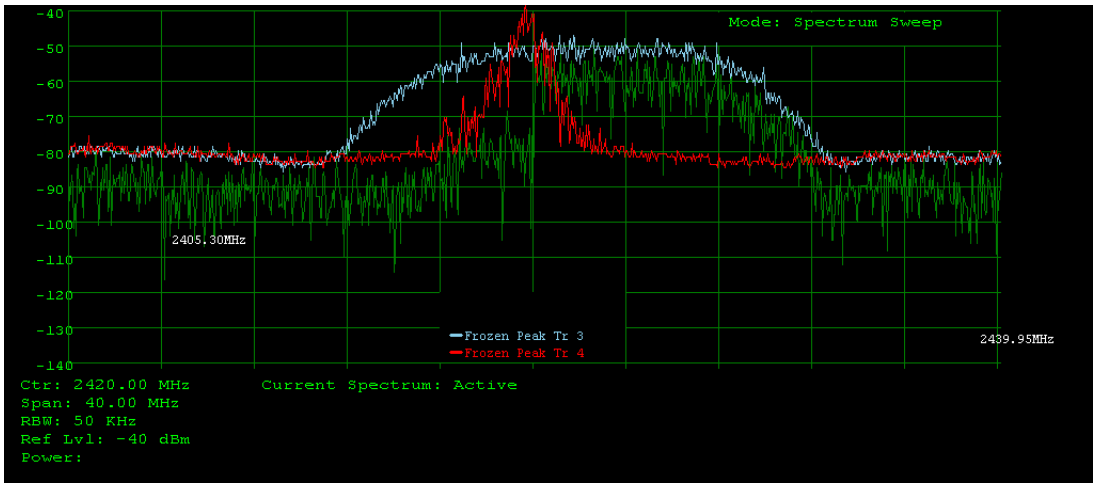


Figure 4.26: Signal traces showing overlap between 802.11 channel 3 (blue) and 802.15.4 channel 14 (red).

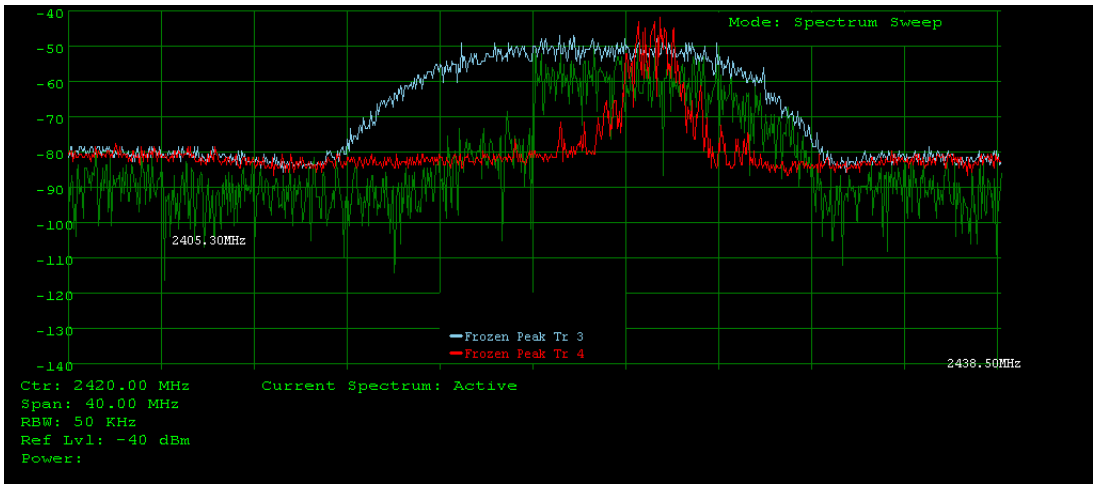


Figure 4.27: Signal traces showing overlap between 802.11 channel 3 (blue) and 802.15.4 channel 15 (red).

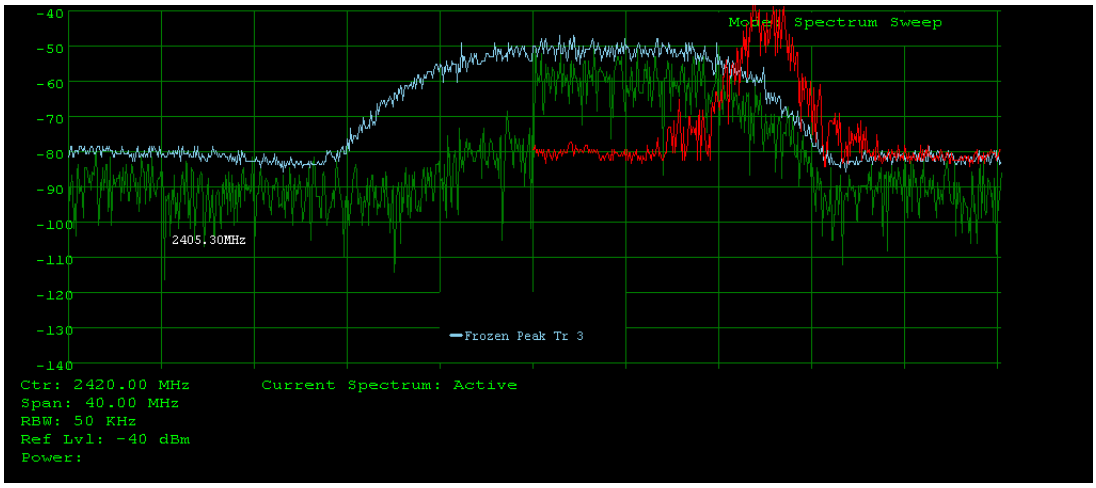


Figure 4.28: Signal traces showing overlap between 802.11 channel 3 (blue) and 802.15.4 channel 16 (red).

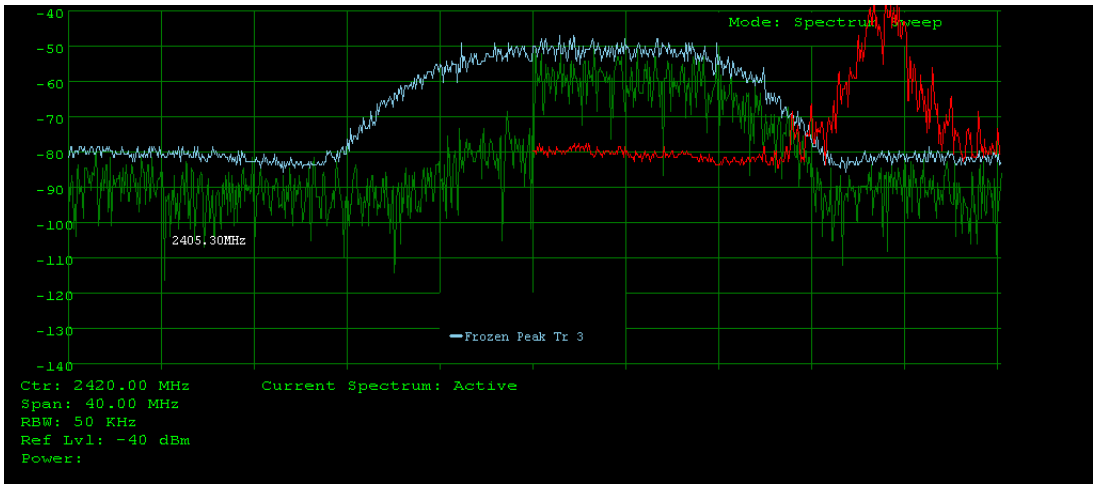


Figure 4.29: Signal traces showing overlap between 802.11 channel 3 (blue) and 802.15.4 channel 17 (red).

4.5 Summary

In this chapter we proposed a novel and generic measurement-based approach for computing the interference factor between two wireless radio channels. Our approach is generic because it is capable of computing the I-factors between channels of different wireless technologies, such as IEEE 802.11 and 802.15.4, as well as between channels of the same wireless technology. We have also proposed two specific measurement-based I-factor definitions and presented their computation methods following a measurement-based approach.

In our wireless testbed, we performed experiments with 802.11b and 802.15.4 radios and computed I-factor values between 802.11 channels and between 802.15.4 and 802.11 channels. We compared our results with the existing results in the literature and we reported new results on ZigBee to Wi-Fi interference. According to our findings, an 802.15.4 radio does not interfere with an 802.11b receiver if their center frequencies are separated by at least 17 MHz. If the center frequencies are separated by 12 MHz, the interference from the 802.15.4 radio on the 802.11 receiver is still negligible. However, if the center frequencies are closer than 12 MHz (for instance when the 802.11 receiver is on Wi-Fi channel 3 and the 802.15.4 interferer is on ZigBee channel 13), the interference from the

802.15.4 radio becomes significant in terms of the 802.15.4 radio's total power (see the third row of Table 4.2).

Chapter 5

Optimization Models for Joint Flow-Radio and Channel Assignment

Having studied the application and network layer performance of multi-hop multi-radio communications on BilMesh in Chapter 3, and having developed quantitative models of adjacent channel interference in Chapter 4, we now turn our attention to the joint flow-radio and channel assignment problem in the context of multi-radio multi-channel WMNs and analyze them first by using the framework of mathematical optimization.

In this chapter, we propose two mathematical optimization models that, unlike previous studies, address the joint flow-radio and channel assignment problem while incorporating the effects of an idealized MAC protocol. We evaluate our models on exemplary network topologies and analyze the relation between link capacity and co-channel and adjacent channel interference. Using the proposed models we also analyze, on small topologies, the prospective advantage of using overlapping channels in addition to orthogonal channels.

In Section 5.1, we discuss our main motivations in developing optimization models for the joint flow-radio and channel assignment problem. In Section 5.2,

we state our system model and objectives, and introduce the formal notation used in this chapter and throughout Chapters 6 and 7. In Section 5.3, we propose our first optimization model which aims to minimize the total interference in the network. In Section 5.4, we propose our second model which aims the minimize the congestion on the busiest link in the network. In Section 5.5, we evaluate and analyze the proposed models using exemplary network topologies. Section 5.6 concludes the chapter.

5.1 Introduction

In Chapter 3, we analyze, through experiments on a testbed, how the physical channel separation between the radios of a multi-radio relay affects the achievable goodput. This chapter is an attempt to develop a set of mathematical tools to analyze interference and capacities of the links exposed to co-channel and adjacent channel interference. We also aim to observe, in the framework of mathematical optimization, the prospective gains (in terms of reduced interference and increased link capacities) of employing overlapping channels for assignment. The insight gained from these models are used to develop centralized and distributed heuristics for the joint flow-radio and channel assignment problem in Chapters 6 and 7.

Our major contribution in this chapter is to develop a mathematical optimization model for the joint flow-radio and channel assignment problem, which is flow-aware and incorporates the effects of a perfect MAC protocol that proportionally shares the available bandwidth among contending transmitters.

5.2 System Model and Objectives

In this chapter, we propose optimization models to decide on flow-radio coupling and to compute the channels to be assigned to radios. We assume that each transmitter uses a fixed power while transmitting a radio signal and that the

wireless medium has only slow-fading characteristics. We also assume traffic sources and destinations are identified a priori in the network, together with the rate of the traffic flowing among them, which can be achieved by traffic monitoring. We assume fixed rate traffic (CBR traffic) and we assume that node positions are fixed and known. Additionally, we assume routing is given a priori, i.e., the end-to-end paths are already known.

Because spatial distances between multi-radio nodes are given a priori and transmission powers are fixed, the problem of minimizing interference is resolved into a joint flow-radio coupling and channel separation optimization problem. We consider overlapping channels while addressing the channel assignment problem and take the flow magnitudes into account. We use the concept of I-factor discussed in Chapter 4 to address adjacent channel interference in our optimization models.

Below we provide our formal notation:

Node definition: A node has D radio interfaces and is denoted by n_i (or just by i where appropriate), where $i \in [1, |N|]$. N denotes the set of multi-radio nodes in the WMN. The k^{th} radio of n_i is denoted by (i, k). The position of n_i, P_i is given as a point in the chosen coordinate system. The transmission range of a node is d_T and its interference range is d_I, where $d_I \geq d_T$.

Flow definition: Because routing (the paths end-to-end flows will follow) is assumed to be given, and end-to-end traffic patterns between node pairs are known a priori (which can also be measured by traffic monitoring), we decompose end-to-end flows into *one-hop unidirectional flows* using the available routing information. Our flow definition is based on these one-hop flows. If multiple end-to-end flows pass through the adjacent nodes i and j in the same direction (e.g. from i to j), then the magnitude of the one-hop unidirectional flow from i to j used in our flow model is the sum of the magnitudes of all of those end-to-end flows. Hence, we consider aggregate flows between two adjacent nodes. Throughout the discussion below, F denotes the set of these one-hop unidirectional (aggregate) flows between neighbor nodes. A flow between nodes i and j (where the definition

Symbol	Meaning
N	Multi-radio node set
F	One-hop flow set
n_i or i	Node i
P_i	Coordinates of n_i
α	Path loss exponent
d_T	Transmission range
d_I	Interference range
D	Radio interface count in a node
(i,k)	k^{th} radio interface of n_i
$f_{i,j,k,l,x}$	Flow from (i,k) to (j,l) on channel x
M	Number of available wireless channels
ρ_{max}	Maximum data rate of a radio

Table 5.1: Definitions of symbols and abbreviations.

of a flow imposes that i and j are one-hop neighbors) is denoted by $f_{i,j,k,l,x}$ or $f_{i,j}$. In the former notation, k is the identification of the radio interface of node i on which the flow is coupled. Similarly, this flow is coupled on the l^{th} radio interface of node j. x denotes that the k^{th} radio of i and the l^{th} radio of j are operating on channel x. This notation is employed in contexts where the channel of the wireless link carrying the flow is relevant. The latter notation just denotes the fact that the flow is between nodes i and j, and the channel of the wireless link carrying the flow is irrelevant. $|f_{i,j}|$ denotes the magnitude of the flow from i to j.

Physical Layer Parameters: The number of available wireless channels is M, for which a typical value for IEEE 802.11b/g is 11 in the Federal Communications Commission (FCC) domain. The maximum data rate of a radio is ρ_{max}, e.g., for an 802.11b radio ρ_{max} is 11 Mbps.

The system model and related notation developed in this section are also used in Chapters 6 and 7. Table 5.1 provides a quick reference for the various symbols used throughout this chapter and the next two chapters.

5.3 Model 1: A Cost Minimization Model Based on Interference Factor

In this section, we propose a cost minimization model for the joint flow-radio and channel assignment problem based on the notion of the interference factor. The model incorporates overlapping channels in addition to orthogonal channels for the assignment problem. The cost definition of the model is as follows:

Cost definition: For a given (one-hop) flow $f_{i,j,k,l,x}$, the cost of the directed arc corresponding to this flow in the network graph (N, F), $c_{i,j,k,l,x}$, is defined as the *total* interference factor caused by a unit transmission from n_i on channel x, computed over the interference set of the transmitter node n_i. The *interference set*, R_i, of n_i is defined as the set of all receiver nodes which have a radio set to an overlapping channel y (y can be equal to x) and are in the interference range (d_I) of n_i. Hence,

$$c_{i,j,k,l,x} = \sum_{(i',j'):\, f_{i',j'} \in F \wedge\, j' \in R_i} \;\; \sum_{y:\, \exists l', \pi^{l'}_{j'} = y} I(x, y), \tag{5.1}$$

where the summation is over all flows $f_{i',j',k',l',y}$ for which $n_{j'}$ is in the interference set of n_i.

An extension on the cost definition is to incorporate slow-fading effects in the I-factor definition:

$$c'_{i,j,k,l,x} = \sum_{(i',j'):\, f_{i',j'} \in F \wedge\, j' \in R_i} \;\; \sum_{y:\, \exists l', \pi^{l'}_{j'} = y} \frac{I(x, y)}{|P_i - P_{j'}|^{\alpha}}, \tag{5.2}$$

where P_i and $P_{j'}$ are the coordinates of n_i and $n_{j'}$, respectively, and α is the path loss exponent of the wireless medium.

The cost formulations outlined above do not take other transmitters in the interference set of a given transmitter into account. The cost model given in (5.3)

incorporates these in-range transmitters by making use of the fact that these transmitters are *receivers* of MAC layer ACK packets. We extend the definition of $c'_{i,j,k,l,x}$ as follows:

$$c''_{i,j,k,l,x} = \sum_{(i',j'):\, f_{i',j'} \in F \wedge\, j' \in R_i} \;\sum_{y:\, \exists l', \pi^{l'}_{j'} = y} \frac{I(x,y)}{|P_i - P_{j'}|^{\alpha}} + \sum_{(i',j'):\, f_{i',j'} \in F \wedge\, i' \in T_i} \;\sum_{y:\, \exists k', \pi^{k'}_{i'} = y} \frac{\omega\, I(x,y)}{|P_i - P_{i'}|^{\alpha}}, \tag{5.3}$$

where T_i is the set of transmitter nodes in the interference range d_I of n_i. ω is the ratio of the MAC layer ACK packet size to the data packet size.

Given these definitions, the joint flow-radio and channel assignment problem can be expressed as a minimization problem as follows:

$$\begin{aligned} &Minimize\ z(\pi) = \sum_{f_{i,j} \in F} c''_{i,j,k,l,x}\ |f_{i,j,k,l,x}|, \\ &Subject\ to: \\ &\bullet\ f_{i,j} \in F \Rightarrow \exists (k,l) : \pi^k_i = \pi^l_j, \\ &\bullet\ \forall (i,k),\ 0 \le \pi^k_i \le M\ \wedge\ 1 \le i \le |N|\ \wedge\ 1 \le k \le D \end{aligned} \tag{5.4}$$

where N is the node set and F is the set of one-hop flows in the network deduced from the traffic patterns together with the routing information. $c''_{i,j,k,l,x}$ is given by (5.3), and $|f_{i,j,k,l,x}|$ is the magnitude of the aggregate one-hop flow emanating from n_i and arriving at n_j. π^k_i denotes the channel which is assigned to the k^{th} radio of n_i ((i,k)). $\pi^k_i = 0$ implies that the radio interface is not employed and $\forall x \in [1, 11]\,,\ I(x,0) = I(0,x) = 0$.

5.4 Model 2: Another Model based on Theoretical Channel Capacities

In this section, we try to model a solution to the joint flow-radio and channel assignment problem using Shannon's channel capacity formula [21] applicable to the AWGN channel:

$$C = B\ lg(1 + \frac{S_R}{N_0 B}), \tag{5.5}$$

where C is the channel capacity (in bps), B is the transmission bandwidth (in Hz), S_R is the received signal power (in W), and $N_0/2$ is the power spectral density (PSD) of the noise (in W/Hz).

The following assumptions are made in addition to the assumptions given in Section 5.2:

1. The radio link can be utilized at its Shannon capacity (given by (5.5)), where $B = 22$ MHz for IEEE 802.11b.
2. The transmit power S is fixed. For calculating S_R, we use a slow-fading only channel model.
3. The adjacent channel and the co-channel interference have both additive white Gaussian noise characteristics.

For a given link (w, z) carrying an aggregate flow $f_{w,z} \in F$ of magnitude $|f_{w,z}|$, we have to determine the total adjacent and co-channel interference added to $N_0 B$, the white Gaussian noise on the channel. The total noise power around receiver node n_z of flow $f_{w,z}$ on channel y, $\eta^y_{w,z}$, is modeled as follows:

$$\eta^y_{w,z} = \left(\sum_{f_{i,j,k,l,x}:\ f_{i,j} \in F \wedge\ z \in R_i} \frac{I(x,y)\ |f_{i,j}|}{|P_i - P_z|^{\alpha}\ C_{w,z,y}} \right) S + N_0 B \tag{5.6}$$

where $C_{w,z,y}$ is the Shannon channel capacity of link (w, z), which is on channel y, modeled after (5.8). N_0B is the AWGN [73] for channel y around receiver node n_z. The factor $\frac{|f_{i,j}|}{C_{w,z,y}}$ is used to weight the average interference power induced on channel y as a function of the magnitude of the interfering flow $f_{i,j}$, which is on channel x. This formulation allows the power from a specific interferer to be scaled with a factor greater than 1.0, because, for a link (i, j) with a higher capacity than link (w, z), it may be the case that $|f_{i,j}| > C_{w,z,y}$.

The Boolean predicate below the summation operator in (5.6) does not take the physical transmitter constraints or MAC constraints into account. When considering the interference power contributed by the flow $f_{i,j,k,l,x}$ nearby the receiver radio of the flow $f_{w,z,k',l',y}$, if both flows are on the same channel (i.e., $x = y$), then we should check whether n_w is in the interference range of n_i or in the transmission range of n_j. In such a case, CSMA/CA and *virtual carrier sensing* will prevent the transmissions of the two flows from occurring simultaneously, hence they will not interfere with each other. Extending (5.6) with this additional constraint yields:

$$\eta^y_{w,z} = \left(\sum_{f_{i,j,k,l,x}:\, f_{i,j} \in F \wedge\, z \in R_i \wedge (x=y \Rightarrow w \notin (R_i \cup R_j))} \frac{I(x,y)\ |f_{i,j}|}{|P_i - P_z|^{\alpha}\ C_{w,z,y}} \right) S + N_0 B \quad (5.7)$$

In (5.7), without loss of generality, we assume that transmission ranges and interference ranges are equal ($d_I = d_T$). The predicate $x = y \Rightarrow w \notin (R_i \cup R_j)$ also makes the physical transmitter constraints irrelevant with the following reasoning:

1. If $x \neq y$, then physical constraints cannot be relevant (because we should have distinct transmitter and receiver radios for these two flows).

2. If $x = y$, then the MAC constraint $w \notin (R_i \cup R_j)$ implies $w \neq i \wedge w \neq j$. And if $z = i \vee z = j$, then $i \in R_w \vee j \in R_w$. But by the symmetry of the links, $w \in (R_i \cup R_j)$, which contradicts with the MAC constraint.

From 5.5 and 5.7, under the aforementioned assumptions, for a given link

(w, z) on channel y, which is to carry the flow $f_{w,z}$, the available Shannon capacity shall be:

$$C_{w,z,y} = B\ lg\left(1 + \frac{S}{|P_w - P_z|^{\alpha}\ \eta^{y}_{w,z}}\right)\ \beta\left(y\right), \tag{5.8}$$

where $\beta\left(y\right)$ models the effects of a perfect MAC protocol. The assumed MAC protocol is perfect in the sense that it perfectly shares the available bandwidth between transmitter radios proportionally with their demands. Hence, under this perfect MAC assumption:

$$\beta\left(y\right) = \frac{|f_{w,z}|}{|f_{w,z}| + \sum_{f_{i,j,k,l,y}:\ f_{i,j} \in F \wedge w \in (R_i \cup R_j)} |f_{i,j}|} \tag{5.9}$$

In (5.9), $w \in R_i$ accounts for the effects of CSMA/CA where the transmitter node n_w's MAC will back-off if it is in the interference set R_i of another transmitter n_i. $w \in R_j$ accounts for the effects of the virtual carrier sensing mechanism (assuming RTS/CTS handshakes are in effect), where the transmitter node n_w hears the CTS packet sent by another receiver node n_j and defers transmission using the information found in the *network allocation vector* (NAV) of the CTS packet. While incorporating the effects of virtual carrier sensing, we assume that the interference range and the transmission range (the range in which a receiver radio can successfully decode messages) are the same regardless of the SNR at the receiver site.

Now, the joint flow-radio and channel assignment problem may be expressed as an optimization problem as follows:

$$\begin{aligned}
&Minimize\ \max_{\forall f_{w,z} \in F}\left\{\frac{|f_{w,z}|}{C_{w,z,y}}\right\}, \\
&Subject\ to: \\
&\bullet\ f_{w,z} \in F \Rightarrow \exists (k,l) : \pi^{k}_{w} = \pi^{l}_{z}, \\
&\bullet\ \forall (w,k),\ 0 \leq \pi^{k}_{w} \leq M\ \ \wedge\ \ 1 \leq w \leq |N|\ \ \wedge\ \ 1 \leq k \leq D,
\end{aligned} \tag{5.10}$$

where $C_{w,z,y}$ is given by (5.8) and $\eta^{y}_{w,z}$ is given by (5.7). The objective of (5.10)

Symbol	Value
B	22 MHz
S	0 dBm
N_0B	-80 dBm
α	2.0
D	2
M	11
d_T	1 m
d_I	1 m
ω	0.01

Table 5.2: Model parameters used for evaluation.

Ch. Separation	I-factor
0	1.0
1	0.61
2	0.33
3	0.14
4	0.09
5	0.03
6	0.01
$7, 8, 9, 10$	0.0

Table 5.3: I-factor values used for evaluation.

is to minimize the maximum link utilization in the network.

5.5 Evaluation

We evaluate the optimization models proposed in Sections 5.3 and 5.4 using four small network topologies for which we can find the optimum solutions by enumeration. The model parameters used in this section are given in Table 5.2. Figures 5.1-5.4 show the optimum solutions of (5.4), and Figures 5.5-5.8 show the optimum solutions of (5.10). The total interference factor function of a flow used in (5.4) is given by (5.3). For these topologies, if $f_{i,j} \in F \Rightarrow |P_i - P_j| = d_T = d_I$.

The I-factor values used in this section are calculated using *SIAM* method of Section 4.2. Table 5.3 lists these values.

Both models yield the same optimal flow-radio couplings and channel configurations for the first three topologies (see Figures 5.1-5.3 and Figures 5.5-5.7). This

Network	Max. Link Utilization	Max. Link Utilization (Constrained)
$\|N\|=3, \|F\|=2$	$1.71\,10^{-4}$	$1.71\,10^{-4}$
$\|N\|=3, \|F\|=3$	$2.87\,10^{-4}$	$2.87\,10^{-4}$
$\|N\|=4, \|F\|=4$	$3.14\,10^{-4}$	$3.42\,10^{-4}$
$\|N\|=6, \|F\|=5$	$5.13\,10^{-3}$	$6.84\,10^{-3}$

Table 5.4: Maximum link utilizations in optimum solutions of (5.10).

implies that with the specific parameters given in Tables 5.2 and 5.3, the flow-radio and channel configuration that yields the minimum total interference also yields the minimum congestion on the busiest wireless link for these exemplary networks. However, as an example, the flow-radio and channel configuration given in Figure 5.8 is not interference-wise optimal (i.e., has the least total interference among feasible configurations).

Another important point to remind is that the equivalence of the optimal solutions of the two models depends on not only the topology but also the model parameters. As an example, when the white noise level drops down to -100 dBm as opposed to the -80 dBm of Table 5.2, the optimal solution of (5.10) for the $|N|=4, |F|=4$ topology changes and is given in Figure 5.7(b), whereas the model in (5.4) does not depend on the noise level. In Figure 5.7(b), the capacity of each link is half the maximum (zero-interference) Shannon capacity (about 365.41 Mbps) because exactly two transmitters equally share the available bandwidth. The corresponding utilization on each link is $2.74\,10^{-4}$. When the white noise power is at -100 dBm for the channel and the (non-optimal) channel configuration $(8, 4, 1, 11)$ is used for the (n_1, n_2), (n_2, n_3), (n_3, n_4) and (n_4, n_1) links respectively, interference from three channels away on the (n_2, n_3) and (n_4, n_1) links decreases their capacities to about 318.41 Mbps, which results in a utilization of $3.14\,10^{-4}$.

In the second column of Table 5.4, we list the maximum link utilizations in the optimum solutions of (5.10) for the topologies given in Figures 5.5-5.8. The utilization of the most congested link gradually increases as the network size increases. In Figure 5.8, the link between n_3 and n_4 is the most congested link, which carries a flow of 2 Mbps.

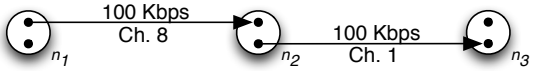


Figure 5.1: Optimum solution of (5.4) for a network where $|N|= 3$, $|F|= 2$.

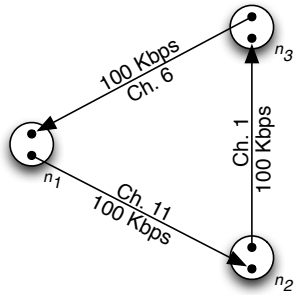


Figure 5.2: Optimum solution of (5.4) for a network where $|N|= 3$, $|F|= 3$.

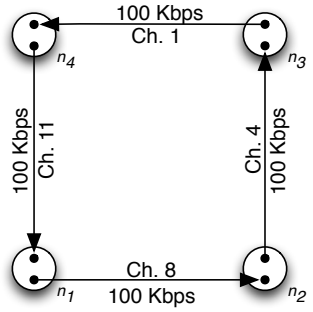


Figure 5.3: Optimum solution of (5.4) for a network where $|N|= 4$, $|F|= 4$.

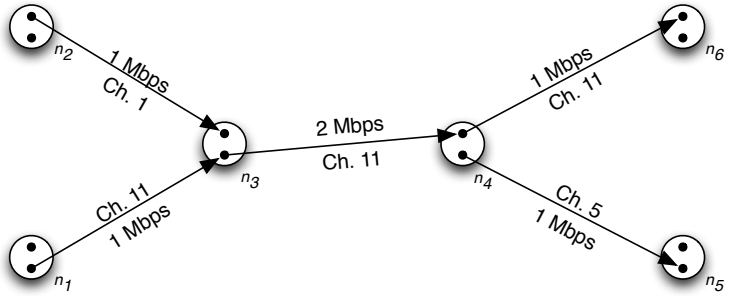


Figure 5.4: Optimum solution of (5.4) for a network where $|N|= 6$, $|F|= 5$.

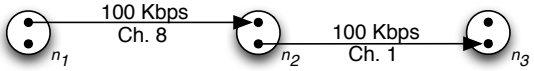


Figure 5.5: Optimum solution of (5.10) for a network where $|N|= 3$, $|F|= 2$.

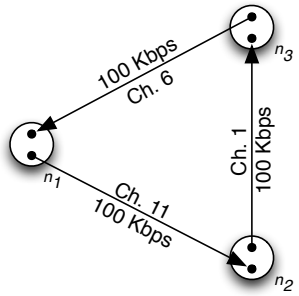
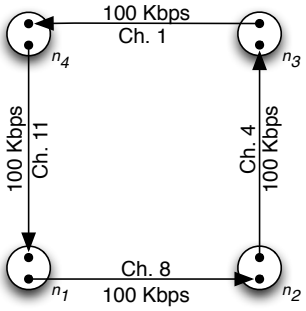
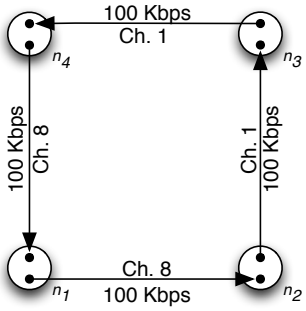


Figure 5.6: Optimum solution of (5.10) for a network where $|N|= 3$, $|F|= 3$.



(a) $N_0B = -80$ dBm



(b) $N_0B = -100$ dBm

Figure 5.7: Optimum solution of (5.10) for a network where $|N|= 4$, $|F|= 4$.

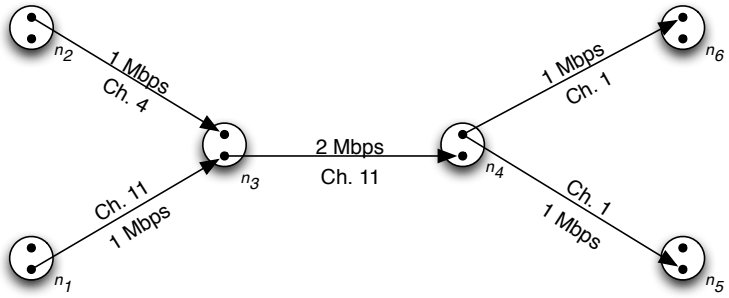


Figure 5.8: Optimum solution of (5.10) for a network where $|N|= 6$, $|F|= 5$.

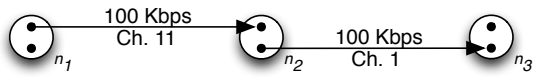


Figure 5.9: Constrained optimum solution of (5.10) for a network where $|N|= 3$, $|F|= 2$.

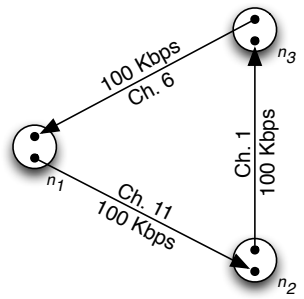


Figure 5.10: Constrained optimum solution of (5.10) for a network where $|N|= 3$, $|F|= 3$.

5.5.1 Constrained Optimization With Three Non-overlapping Channels

In this section, we solve the optimization problem given as (5.10) by using a constrained channel space consisting only of channels 1, 6 and 11. These three channels are commonly considered as the *non-overlapping* channels of IEEE 802.11b/g. Figures 5.9-5.12 show the constrained optimum solutions for the five small topologies examined above. The model parameters used here are the same as in Table 5.2 with the exception of $M = 3$.

In the third column of Table 5.4, we list the maximum link utilizations in the constrained optimum solutions of (5.10) for the topologies given in Figures 5.9-5.12. As revealed in this table, the unconstrained solver (where $M = 11$) achieves lower maximum congestion values than the constrained solver, if it can utilize overlapping channels in addition to orthogonal channels (see Figures 5.7 and 5.8, where $|N|= 4, |F|= 4$ and $|N|= 6, |F|= 5$, respectively).

5.5.2 Congestion As a Function of Distance

We now turn our attention to how the utilization of the maximally congested link changes as a function of the distance between the link's endpoints. For studying the effects of link distance on the optimization model (5.10), the topology given

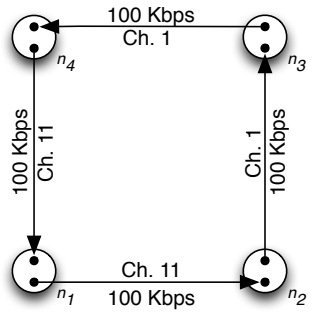


Figure 5.11: Constrained optimum solution of (5.10) for a network where $|N|= 4$, $|F|= 4$.

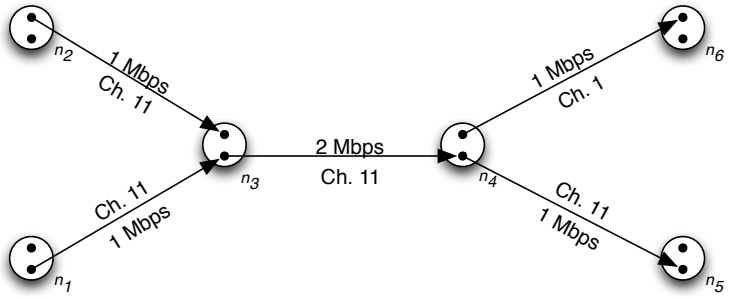


Figure 5.12: Constrained optimum solution of (5.10) for a network where $|N|= 6$, $|F|= 5$.

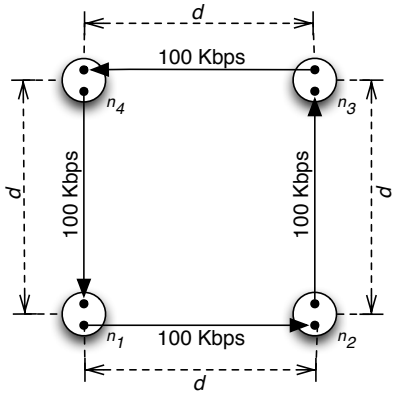


Figure 5.13: Network of 4 nodes and 4 flows and the distance parameter d.

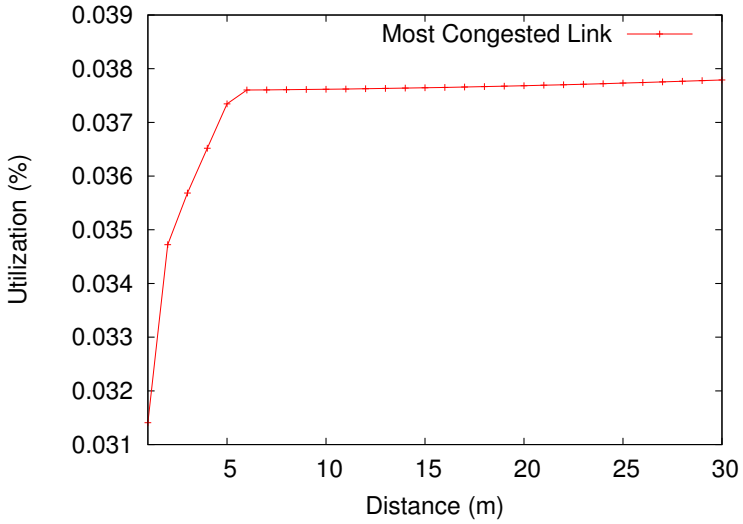


Figure 5.14: Utilization of the most congested link in the optimal solutions of (5.10) for the topology in Figure 5.13.

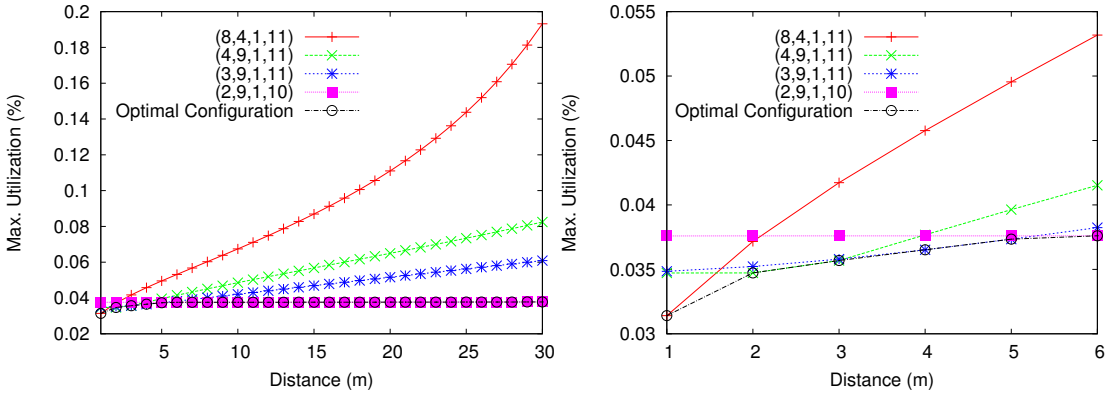
Interval	**Channels**
$d = 1$	$(8, 4, 1, 11)$
$2 \leq d \leq 3$	$(4, 9, 1, 11)$
$4 \leq d \leq 5$	$(3, 9, 1, 11)$
$6 \leq d \leq 30$	$(2, 9, 1, 10)$

Table 5.5: Optimum channel configurations for specific intervals of d.

in Figure 5.13 is used.

Figure 5.14 shows the relation between the utilization of the most congested link in the optimal solution and the internodal distances (d) in the topology of Figure 5.13. Here, the interference range (d_I) and the transmission range (d_T) of each multi-radio node is taken as d, which is varied from 1 m to 30 m. As the internodal distances increase and the transmission power being kept constant, the capacities of the links decrease. As a result, the utilization of the most congested link also increases. Furthermore, the optimal channel configuration changes: As the internodal distances increase, it becomes gradually more critical to spectrally separate the neighbor links (links incident on a common node) than to separate the links that are not neighbors. Table 5.5 shows the channel configurations that yield optimal solutions of (5.10) for specific intervals of d.

Figure 5.15(a) shows the plots of the utilizations of the most congested links



(a) $d \in [1, 30]$ (b) $d \in [1, 6]$

Figure 5.15: Utilizations of the most congested links for the topology in Figure 5.13 with fixed and optimal channel configurations.

of the topology in Figure 5.13 for $d \in [1, 30]$, with *fixed* channel configurations of $(8, 4, 1, 11)$, $(4, 9, 1, 11)$, $(3, 9, 1, 11)$, $(2, 9, 1, 10)$, and the utilization of the most congested link in the optimal solution. As the figure reveals, a configuration optimal in an interval of d may no longer be optimal in another disjoint interval. Furthermore, there is no single channel configuration which yields an optimal solution in the interval $[1, 30]$ of d. Figure 5.15(b) plots the same data for the interval $[1, 6]$ to clarify how the optimal channel configuration changes in this interval.

In Figure 5.16, the utilizations of the most congested links in the constrained optimal solutions of (5.10) are shown together with the unconstrained optimal solutions for the topology in Figure 5.13. The unconstrained optimal solutions employ overlapping channels in addition to orthogonal channels, whereas the constrained solutions are limited to orthogonal channels of IEEE 802.11b. Using overlapping channels for assignment can decrease the utilization of the most heavily congested link down to 70% compared to using only orthogonal channels.

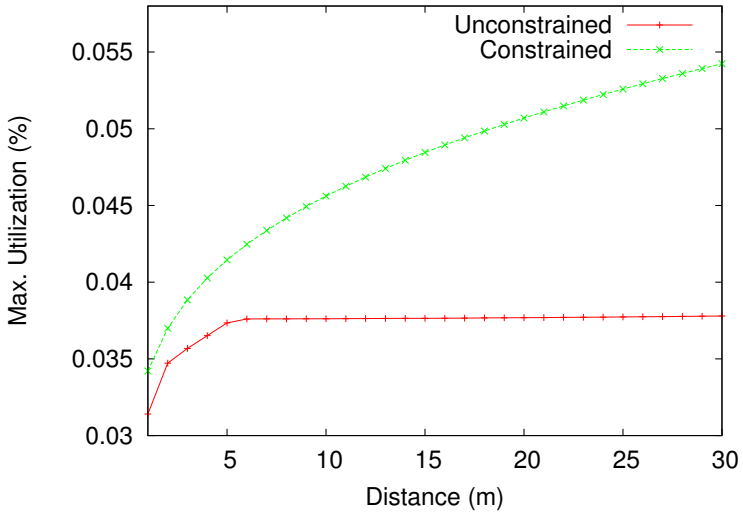


Figure 5.16: Utilizations of the most congested links in unconstrained and constrained optimal solutions of (5.10) for the topology in Figure 5.13.

5.6 Summary

In this section, we have proposed two optimization models for the joint flow-radio and channel assignment problem in the context of multi-radio multi-channel WMNs. Unlike the majority of the literature on channel assignment, both of our models make use of overlapping channels in addition to orthogonal channels.

The first model we propose (given as (5.4)) incorporates the effects of a CSMA/CA like MAC protocol through the ω parameter, which is defined as the ratio of the MAC layer ACK packet size to the data packet size. Unlike previous work, this allows our model to treat flow sources as receivers of MAC layer ACK packets. In the second model (given as (5.10)), we take the effects of a *perfect* MAC protocol into account through the $\beta(y)$ function, which shares the available bandwidth proportionally among contending transmitters.

We evaluate the proposed models by enumerating all feasible solutions on small topologies of two, three, four and six two-radio nodes. For our exemplary four node topology, using overlapping channels in addition to orthogonal ones decreases the congestion on the busiest link down to 70%, compared to using only orthogonal channels of IEEE 802.11b/g when the nodes are separated by 30 m.

With our current solvers, which basically enumerate the solutions to pick the optimal, it is infeasible to solve the models for larger topologies with greater numbers of nodes, flows, or radios per node. Mixed integer linear programming models that incorporate the effects of MAC protocols, similar to the models proposed in this section, can be developed. Then, solvers like CPLEX [74] can be used to solve these MILP models.

It is also possible to develop mathematical models on the joint flow-radio and channel assignment problem in the context of Adaptive Modulation and Coding (AMC) systems [75], which consider the available modulation and coding schemes in a wireless communication standard (such as the IEEE 802.11b/g/n or IEEE 802.16) as optimization parameters.

Chapter 6

Centralized Algorithms for Joint Flow-Radio and Channel Assignment Using Partially Overlapping Channels in Multi-Radio Wireless Mesh Networks

Equipping mesh nodes with multiple radios that support multiple wireless channels is considered a promising solution to overcome the capacity limitation of single-radio wireless mesh networks. However, careful and intelligent radio resource management is needed to take full advantage of the extra radios on the mesh nodes. Flow-radio assignment and channel assignment procedures should obey the physical constraints imposed by the radios (see Section 6.1) as well as the topological constraints imposed by routing. Varying numbers of wireless channels are available for the channel assignment procedure for different wireless communication standards. To further complicate the problem, the wireless communication standard implemented by the radios of the wireless mesh network

may define overlapping as well as orthogonal channels, as in the case of the IEEE 802.11b/g family of standards.

In this chapter, we present two centralized schemes that address the joint flow-radio and channel assignment problem and evaluate their performance on random topologies. In Section 6.1, we discuss our motivations, list our main contributions, and define the problem scope for this chapter. In Section 6.2, we propose two centralized schemes for joint flow-radio and channel assignment. In Section 6.3, we introduce the metrics we use for performance evaluation and discuss our experiment results. Section 6.4 concludes the chapter.

6.1 Introduction

As discussed in Section 2.3, most existing studies on channel assignment in the context of multi-radio WMNs consider only non-overlapping (orthogonal) channels. However, as previous research [23–25, 41] has shown, using partially overlapping as well as orthogonal channels for channel assignment better utilizes the spectrum and can increase the overall capacity and aggregate throughput in the WMN by mitigating interference.

When addressing the channel assignment problem, we consider the following physical constraints of the radios:

1. Two radios must be tuned to the same wireless channel to efficiently communicate with each other.
2. The total (inbound and outbound) traffic on a radio cannot exceed the maximum data rate (ρ_{max}) of that radio.

Because of the physical constraints of the radio interfaces, two multi-radio neighbor nodes (nodes in each other's communication range) must have at least one of their radios tuned to a common channel to communicate with each other. Otherwise, there will be no wireless link established between these neighbors.

To complicate the situation further, the decision of which radio of a (multi-radio) node will be used to communicate with its neighbor (flow to radio assignment) has a substantial effect on the NP-hard channel assignment. Despite its importance, due to the complexity of the channel assignment problem, the flow-radio assignment problem is overlooked in most existing studies.

In this chapter, we propose two centralized schemes within a unified framework that address the flow-radio assignment and channel assignment problems in a joint manner. The main contributions of this chapter are as follows:

- We address the joint flow-radio and channel assignment problem with centralized heuristics.
- Unlike most existing studies, we consider overlapping as well as orthogonal channels for channel assignment.
- We propose novel, realistic and easy to compute metrics for assessing the interference and the residual capacities of the receivers.

This chapter shares the system model description and underlying assumptions given in Section 5.2, with the exception that our aim in this chapter is to develop centralized algorithms to decide on flow-radio assignment and to compute the channels to be assigned to radios. The same set of symbols given in Table 5.1 is used in this chapter with the addition of two new symbols: O_{Δ} for denoting the channel separation between two consecutive orthogonal channels (e.g., $O_{\Delta} = 5$ for 802.11b/g), and N_d for denoting the average node degree of a random network topology (see Section 6.3).

Table 5.1 is replicated here, as Table 6.1, for ease of reference together with the addition of new symbols.

Symbol	Meaning
N	Multi-radio node set
F	One-hop flow set
n_i or i	Node i
P_i	Coordinates of n_i
α	Path loss exponent
d_T	Transmission range
d_I	Interference range
D	Radio interface count in a node
(i,k)	k^{th} radio interface of n_i
$f_{i,j,k,l,x}$	Flow from (i,k) to (j,l) on channel x
M	Number of available wireless channels
ρ_{max}	Maximum data rate of a radio
O_Δ	Orthogonal channel separation
N_d	Avg. node degree for a random topology

Table 6.1: Definitions of symbols and abbreviations.

6.2 Centralized Algorithms for Joint Flow-Radio and Channel Assignment

6.2.1 Flow-Radio Assignment

A multi-radio mesh node is normally free to assign each aggregated single-hop flow entering or exiting itself to one of its radios independently without having to consider other nodes. Our first flow-radio assignment heuristic evenly distributes the total traffic (inbound and outbound) among the radios of a multi-radio node, so that the flows will have a greater chance of being assigned to different channels, reducing co-channel interference. This also promotes the higher utilization of the available radio resources of a node and increases available capacity in the WMN. Algorithm 5 outlines this heuristic for flow-radio assignment subproblem. The algorithm is executed for each multi-radio node in the network.

Algorithm 5 tends to leave flows with relatively large bandwidth demands on their own radios and in this way, gives the channel assignment procedure a chance to decouple relatively high traffic flows, reducing interference. It also treats (aggregated) flows as atoms, meaning that it does not divide a flow among multiple radios in a node. This approach ensures that all packets belonging to the same single-hop flow are transmitted and received by the same radios respectively at

Algorithm 5 Flow-Radio Assignment for n_i
Input: F
Output: $C_i : F \rightarrow R$, flow-radio assignment information for n_i

1: **procedure** FRASSIGN(F)
2: $S_k \leftarrow 0, \forall k \in [1, D]$ ▷ S_k is the total traffic coupled on radio (i, k)
3: $T \leftarrow$ Sorted array of inbound and outbound flows f, in non-increasing order using their bandwidth demands as keys
4: **for** $m = 1$ to length[T] **do**
5: Select k such that S_k is minimum in S
6: $C_i[f_m] \leftarrow k$
7: $S_k \leftarrow S_k +$ key[T_m]
8: **end for**
9: FLOWBALANCER(C_i)
10: **end procedure**

the sending and receiving nodes. This is also true for a multi-hop flow, which is decomposed into single-hop flows by the single-path routing protocol. Hence, Algorithm 5 ensures that all packets of a multi-hop flow experience similar channel conditions in exactly the same order, although they may be transmitted on different wireless channels and at varying levels of interference. This method mitigates packet reordering problems that adversely affect the performance of reliable transport protocols or real-time applications.

If there is bidirectional traffic between two neighbor nodes, to further reduce interference, the (one-hop) flows between these neighbors can be assigned to the same radios. The heuristic given with Algorithm 6 assigns all flows between two nodes to the same radios on these nodes (see Line 17), as long as the capacity constraints of the radios are not violated. In Algorithm 6, src[f] and dst[f] denote the source and the destination nodes of the flow f, respectively. A flow from node i to j and another flow from j to i are coupled on the same radios of i and j. Hence, Algorithm 6 concentrates all flows between two neighboring nodes on the same radios.

Because both of Algorithms 5 and 6 are heuristics that schedule all packets of a (single-hop) flow on the same transmitter and receiver, they may fail to find a feasible schedule. The heuristic we propose with Algorithm 7 tries to balance a node's overflown and underflown radios if capacity constraints are violated.

Algorithm 6 Concentrating Flow-Radio Assignment for n_i
Input: F
Output: $C_i : F \rightarrow R$, flow-radio assignment information for n_i

1: **procedure** CFRAssign(F)
2: $S_k \leftarrow 0, \forall k \in [1, D]$ ▷ S_k is the total traffic coupled on radio (i, k)
3: $T \leftarrow$ Sorted array of inbound and outbound flows f, in non-increasing order using their bandwidth demands as keys
4: $c_m \leftarrow False,\ \forall m \in$ [1,length[T]] ▷ c_m is true if and only if the flow f_m has been assigned to some radio on node i
5: **for** $m = 1$ to length[T] **do**
6: **if** $c_m = True$ **then**
7: `continue`
8: **end if**
9: Select k such that S_k is minimum in S
10: $C_i[f_m] \leftarrow k$ ▷ Couple f_m with (i, k)
11: $c_m \leftarrow True$
12: $S_k \leftarrow S_k +$ key[T_m]
13: **for** $n = m + 1$ to length[T] **do**
14: **if** $S_k + key[T_n] > \rho_{max}$ **then**
15: `continue`
16: **end if**
17: **if** (src[f_m] $= i \wedge$ dst[f_m] = src[f_n]) $\vee$ (dst[f_m] $= i \wedge$ src[f_m] = dst[f_n]) **then**
18: $C_i[f_n] \leftarrow k$ ▷ Assign f_n to (i, k)
19: $c_n \leftarrow True$
20: $S_k \leftarrow S_k + key[T_n]$
21: **end if**
22: **end for**
23: **end for**
24: FlowBalancer(C_i)
25: **end procedure**

Algorithm 7 Flow Balancer on n_i
Input: C_i
Output: C_i

procedure FLOWBALANCER(C_i)

$$\Omega \leftarrow \left\{ (i,k) : \; k \in [1,D] \wedge \sum_{\forall (j,l): f_{i,j,k,l} \in F} |f_{i,j,k,l}| + \sum_{\forall (j,l): f_{j,i,l,k} \in F} |f_{j,i,l,k}| > \rho_{max} \right\}$$

▷ Ω is the set of overflown radios

$$\Upsilon \leftarrow \left\{ (i,k) : \; k \in [1,D] \wedge \sum_{\forall (j,l): f_{i,j,k,l} \in F} |f_{i,j,k,l}| + \sum_{\forall (j,l): f_{j,i,l,k} \in F} |f_{j,i,l,k}| < \rho_{max} \right\}$$

▷ Υ is the set of underflown radios, i.e. radios with positive residual capacity

for all $(i,k) \in \Omega$ **do**
  $E_k \leftarrow \sum_{\forall (j,l): f_{i,j,k,l} \in F} |f_{i,j,k,l}| + \sum_{\forall (j,l): f_{j,i,l,k} \in F} |f_{j,i,l,k}| - \rho_{max}$
  for all $f_{R_k} : f_{i,j,k,l} \in F \vee f_{j,i,l,k} \in F$ **do** ▷ For each flow on (i,k)
    for all $(i,k') \in \Upsilon$ **do**
      for all $f_{R_{k'}} : f_{i,j,k',l} \in F \vee f_{j,i,l,k'} \in F$ **do** ▷ For each flow on (i,k')
        $\delta \leftarrow |f_{R_k}| - \left|f_{R_{k'}}\right|$
        $\Delta_{R_{k'}} \leftarrow \rho_{max} - \sum_{\forall (j,l): f_{i,j,k',l} \in F} |f_{i,j,k',l}| - \sum_{\forall (j,l): f_{j,i,l,k'} \in F} |f_{j,i,l,k'}|$
        if $\delta > 0 \wedge E_k - \delta \geq 0 \wedge \Delta_{R_{k'}} - \delta \geq 0$ **then**
          $E_k \leftarrow E_k - \delta$
          $C_i[f_{R_{k'}}] \leftarrow k$
          $C_i[f_{R_k}] \leftarrow k'$
        end if
      end for
    end for
  end for
  if $E_k > 0$ **then**
    `exit` ▷ Heuristic failed to balance
  end if
end for
end procedure

6.2.2 Channel Assignment to Interfaces

A node is not free to assign arbitrary channels to its radio interfaces because it communicates with neighbor nodes via these interfaces and two communicating radios must be tuned to the same channel. Hence, the channel assignment procedure has to be coordinated in either a centralized or a decentralized manner so that two nodes which share flows both assign the same channel to their radio interfaces on which these flows are coupled. This will ensure that a wireless link is established between nodes i and j if flow $f_{i,j}$ exists in the network. In Algorithm 8, we propose a centralized channel assignment heuristic. Our channel assignment is static, meaning that once channels are set, they are not changed until the next network-wide assignment which can happen periodically or when traffic conditions change significantly. In Chapter 7, we propose a distributed scheme which addresses the joint flow-radio and channel assignment problem.

Algorithm 8 Centralized Channel Assignment
Input: N, node set
Input: $C : F \rightarrow R$, flow-radio assignment information
Output: π, selected channel table for the radios of the network (N, F)

```
1:  procedure CCA(N, C)
2:    π_i^k ← 0, ∀i ∈ N ∧ ∀k ∈ [1, D]
3:    M ← Sorted array of nodes in N, in non-increasing order using the total inbound and
      outbound traffic on a node as its key
4:    for i = 1 to length[M] do
5:      R ← Sorted array (of length D) of the radios of node M_i in non-increasing order using
      the total inbound and outbound traffic on the interface as its key
6:      for k = 1 to D do
7:        if (i, k) does not have any flows coupled with it then
8:          continue
9:        else if (i, k) has already been assigned a channel then
10:         continue
11:       else                          ▷ M_i is free to select a new channel for its radio (i, k)
12:         SICA(π, (i, k))
13:       end if
14:       if A channel has been allocated for (i, k) then
15:         Assign the same channel to all radios connected with (i, k) in (N, F)
16:       end if
17:     end for
18:   end for
19: end procedure
```

Algorithm 8 processes nodes in N in non-increasing order of their total inbound and outbound traffic, to leave heavily loaded nodes free to select channels

for their interfaces. As proposed in Algorithm 9, nodes try to keep self-interference at a minimum, hopefully decreasing the total interference in the network. We call this approach as Self-Interference minimizing Channel Assignment (*SICA*).

Algorithm 9 Channel Assignment to (i, k) Considering Self-interference Only
Input: π, selected channel table
Input: (i, k), radio for which a channel is to be selected
Output: π_i^k, the channel selected for (i, k)

1: **procedure** SICA($\pi, (i, k)$)
2: $A_i \leftarrow \left\{(i, l) : \pi_i^l \neq 0 \ \wedge \ 1 \leq l \leq D\right\}$ ▷ A_i is the set of all radios of n_i already assigned a channel
3: $\sigma_c \leftarrow 0, \forall c \in [1, M]$ ▷ σ_c is an indication of the total (self-)interference on channel c
4: **for all** $(i, l) \in A_i$ **do**
5: **for** $c = 1$ to M **do**
6: $\sigma_c \leftarrow \sigma_c + I(\pi_i^l, c) \frac{x_{i,l}}{\rho_{max}}$ ▷ Where $x_{i,l}$ is the total inbound and outbound traffic on (i, l) and ρ_{max} is the maximum data rate of (i, l)
7: **end for**
8: **end for**
9: $\pi_i^k \leftarrow c$ such that σ_c is minimum in σ
10: **end procedure**

Algorithm 9 tries to minimize self-interference of each multi-radio node (i.e., interference between a node's own radios) and hence can be carried out in a distributed manner. $\frac{x_{i,l}}{\rho_{max}}$ is used to take the traffic load of a radio into account when selecting channels, where $x_{i,l}$ is the total inbound and outbound traffic on (i, l), and ρ_{max} is the maximum data rate of a radio.

Algorithm 10 is another heuristic we propose for selecting a channel for the radio (i, k). In this algorithm, we do not only consider self-interference but we also take interference from other transmitters in the network into account when selecting a channel for radio (i, k). We call this approach as Global Interference Channel Assignment (*GICA*). To use the GICA scheme instead of SICA, Line 12 of Algorithm 8 is replaced with a call to Algorithm 10.

With Algorithm 10, as each multi-radio node is visited by Algorithm 8 in non-increasing order of their total traffic, the channel that will have the least total utilization weighted protocol interference on radios already assigned a channel is selected for the radio under consideration.

Algorithm 10 Channel Assignment to (i,k) Considering Global Interference
Input: π, selected channel table
Input: (i,k), radio for which a channel is to be selected
Output: π_i^k, the channel selected for (i,k)

procedure GICA($\pi,(i,k)$)
  $A \leftarrow \{(j,l):\ \pi_j^l \neq 0\ \wedge\ 1 \leq j \leq |N| \wedge 1 \leq l \leq D\}$ ▷ A is the set of all radios (of all nodes) in the network that have so far been assigned a channel and thus, $(i,k) \notin A$
  $\sigma_c \leftarrow 0, \forall c \in [1,M]$ ▷ σ_c is an indication of the total (global-)interference on channel c
  for all $(j,l) \in A$ **do**
    for $c = 1$ to M **do**
      $\sigma_c \leftarrow \sigma_c + I(\pi_j^l, c)\ \frac{x_{j,l}}{\rho_{max}}$ ▷ Where $x_{j,l}$ is the total inbound and outbound traffic on (j,l) and ρ_{max} is the maximum data rate of (j,l)
    end for
  end for
  $\pi_i^k \leftarrow c$ such that σ_c is minimum in σ
end procedure

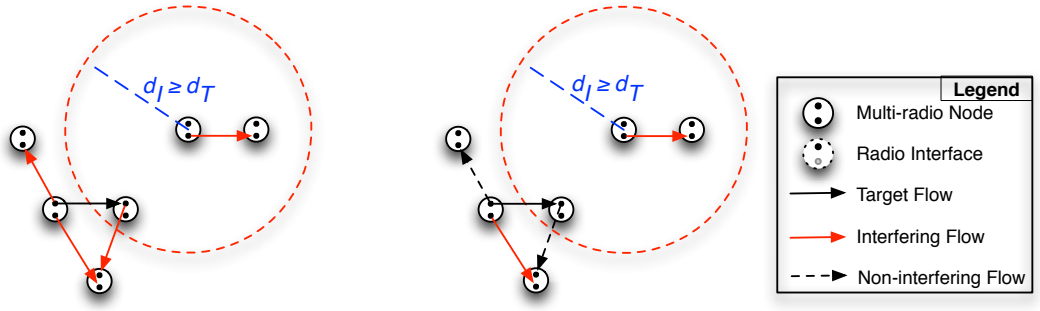
6.3 Evaluation

In this section, we first introduce and discuss in detail the metrics we use for assessing the performance of the proposed algorithms. Then we discuss our experimentation methodology and present our simulation experiment results.

6.3.1 Evaluation Metrics

In this section, we propose and describe in detail some metrics that can be used in evaluating the performance of channel assignment schemes. We used these metrics in evaluating the effectiveness and performance of our proposed centralized joint flow-radio and channel assignment algorithms. These metrics are computed for random and single-channel assignment schemes (configurations), as well as for our SICA and GICA configurations, to assess the performance of the proposed algorithms in mitigating interference and in increasing residual capacity. The first of the proposed metrics, *average protocol interference*, is also used to determine whether a given flow-radio coupling and channel configuration is feasible. These metrics are also used in the evaluation of our distributed scheme proposed in the next chapter.

6.3.1.1 Average Protocol Interference Metric



(a) Typical interference scenarios in the context of I_{ap}.

(b) Typical interference scenarios in the contexts of I_{awp} and R_{bc}.

Figure 6.1: Typical interference scenarios in the contexts of the evaluation metrics.

The average protocol interference metric, I_{ap}, uses the concept of the I-factor [25], assuming a constant transmission power for each transmitter radio. This metric can use any I-factor model [24, 25, 68], $I(x, y)$, where $I(x, y)$ is the normalized amount of interference signal power a transmitter operating on channel x puts on a receiver operating on channel y. In this metric, we do not take the effects of *slow-fading* [2] into account and assume that a *constant* fraction of the transmission power leaks on adjacent channels (defined by the I-factor model in use) throughout the interference range of a transmitter radio (we are concerned about protocol interference). Outside the interference range of the transmitter, the interference power becomes 0 (i.e., no interference). I_{ap} is calculated as follows for a given network (N, F), where N is the set of multi-radio nodes and F is the set of (one-hop) flows:

$$F_{f_{i,j,k,l,x}} = \left\{ \begin{array}{l} f_{i',j',k',l',y} : f_{i',j',k',l',y} \in F \wedge \\ f_{i',j',k',l'} \neq f_{i,j,k,l} \wedge \\ |P_j - P_{i'}| \leq d_I \end{array} \right\}$$

$$i_f((j,l),x) = \sum_{f_{i,j,k,l,x} \in F} \sum_{f_{i',j',k',l',y} \in F_{f_{i,j,k,l,x}}} I(x,y) \tag{6.1}$$

$$I_{ap} = \frac{\sum_{(j,l,x):\exists(i,k), f_{i,j,k,l,x} \in F} i_f((j,l),x)}{|\{(j,l) : \exists(i,k), f_{i,j,k,l} \in F\}|}.$$

In (6.1), $F_{f_{i,j,k,l,x}}$ is the set of flows whose transmitters interfere with the receiver (j,l) of the flow $f_{i,j,k,l,x}$. The definition of $F_{f_{i,j,k,l,x}}$ implies full-duplex operation of the radios. Typical interference scenarios captured by $F_{f_{i,j,k,l,x}}$ are depicted in Figure 6.1(a), where the target flow in the figure corresponds to $f_{i,j,k,l,x}$. The $i_f((j,l),x)$ value is the *total* protocol interference on (j,l), which operates on channel x. The metric, I_{ap}, quantifies the average protocol interference on the receiver radios in the network.

6.3.1.2 Average Physical Interference Metric

The average physical interference metric, I_{aph}, is similar to I_{ap} but takes slow-fading into account while calculating the interference on a receiver from an interferer. More precisely, I_{aph} is given by:

$$i_f((j,l),x) = \sum_{f_{i,j,k,l,x} \in F} \sum_{f_{i',j',k',l',y} \in F_{f_{i,j,k,l,x}}} \frac{I(x,y)}{|P_j - P_{i'}|^\alpha}$$

$$I_{aph} = \frac{\sum_{(j,l,x):\exists(i,k), f_{i,j,k,l,x} \in F} i_f((j,l),x)}{|\{(j,l) : \exists(i,k), f_{i,j,k,l} \in F\}|}, \tag{6.2}$$

where the definition of $F_{f_{i,j,k,l,x}}$ is given in (6.1).

6.3.1.3 Average Weighted Protocol Interference Metric

This metric aims to quantify the average of the flow-magnitude weighted protocol interference over all receiver radios in the network. The I-factor is again used to quantify the amount of interference between wireless channels. The average weighted protocol interference, I_{awp}, for a given network (N, F) is defined as follows:

$$H_{f_{i,j,k,l,x}} = \left\{ \begin{array}{l} f_{i',j',k',l',y} : f_{i',j',k',l',y} \in F \wedge \\ (i',k') \neq (j,l) \wedge (i',k') \neq (i,k) \wedge \\ |P_j - P_{i'}| \leq d_I \end{array} \right\}$$

$$i_w((j,l),x) = \sum_{f_{i,j,k,l,x} \in F} \sum_{\substack{f_{i',j',k',l',y} \in \\ H_{f_{i,j,k,l,x}}}} \frac{|f_{i',j',k',l',y}|}{\rho_{max}} I(x,y) \tag{6.3}$$

$$I_{awp} = \frac{\sum_{(j,l,x):\exists(i,k), f_{i,j,k,l,x} \in F} i_w((j,l),x)}{|\{(j,l) : \exists(i,k), f_{i,j,k,l} \in F\}|}.$$

In (6.3), d_I is a transmitter's interference range. $H_{f_{i,j,k,l,x}}$ is the set of flows whose transmitters interfere with the receiver (j,l) of the flow $f_{i,j,k,l,x}$. $i_w((j,l),x)$ is the *total* weighted protocol interference on (j,l), which operates on channel x. To calculate $i_w((j,l),x)$, we consider the following rules:

- A transmission from (i,k) does not interfere with the receivers of other transmissions of (i,k).
- The transmissions from (j,l) does not interfere with the receptions on (j,l).

These rules correspond to the half-duplex operation of the radios in the network. Typical interference scenarios captured by $H_{f_{i,j,k,l,x}}$ are depicted in Figure 6.1(b), where the target flow in the figure corresponds to $f_{i,j,k,l,x}$. I_{awp} also takes into account that a high traffic flow will have greater interference on a given receiver than a lower traffic flow on the same channel as itself. To capture this fact within I_{awp}, the interference from the transmitter of a given flow on a given receiver is weighted with the normalized flow time. The maximum interference that can be put by an interferer on a receiver is 1, which will be the case if the interferer is:

- operating on the same channel as the receiver under consideration
- and transmitting a total traffic of ρ_{max} bps, fully utilizing its capacity, which implies that the interferer receives no data itself.

6.3.1.4 Receiver Binary Capacity Model and Average Residual Capacity Metric

Next we propose an average residual capacity metric for the receiver radios in the network that is closely related to the total amount of interference in the network. As the interference on a receiver increases, the residual capacity on that receiver decreases. Hence, a good scheme that performs intelligent channel planning in a WMN should utilize radios and increase their residual capacities. To define the average residual capacity metric, R_{bc}, for a given flow-radio coupling and channel configuration, we first define our *binary capacity* model, BC, for a given receiver (j, l) operating on channel x as follows:

$$i_h((j,l),x) = \sum_{(i',k',y)} \sum_{\substack{f_{i,j,k,l,x} \in F \wedge \\ f_{i',j',k',l',y} \in H_{f_{i,j,k,l,x}}}} I(x,y) \tag{6.4}$$

$$BC_{(j,l),x} = \begin{cases} 0 & \text{if } i_h((j,l),x) \geq I_{thres} \\ \rho_{max} & \text{Otherwise} \end{cases}.$$

In (6.4), $H_{f_{i,j,k,l,x}}$ is the set of interferer flows of $f_{i,j,k,l,x}$, as given in (6.3). The total protocol interference on (j,l), operating on channel x, is given as $i_h((j,l),x)$. The binary capacity model assumes that if the total protocol interference on a receiver is above a specified threshold, I_{thres}, then the capacity of that receiver is 0 and no reception is possible. When $I_{thres} = 1$, an interferer operating on the same wireless channel as a receiver will make it impossible for that receiver to receive and correctly decode any data.

Having defined the binary capacity model, the average residual capacity metric, R_{bc}, is given as follows:

$$\Delta_{(j,l),x} = \left(BC_{(j,l),x} - \sum_{f_{i',j,k',l} \in F} |f_{i',j,k',l}| - \sum_{f_{j,i',l,k'} \in F} |f_{j,i',l,k'}| \right) \tag{6.5}$$

$$R_{bc} = \frac{\sum_{(j,l,x):\exists(i,k), f_{i,j,k,l,x} \in F \wedge \Delta_{(j,l),x} \geq 0} \Delta_{(j,l),x}}{|\{(j,l) : \exists(i,k), f_{i,j,k,l} \in F\}|}.$$

In (6.5), $\Delta_{(j,l),x}$ denotes the residual binary capacity of the receiver (j,l), which is on channel x. R_{bc} is the average of the residual capacities of the receiver radios with non-negative residual capacities.

6.3.2 Experiments

We performed extensive simulation experiments to evaluate the flow-radio and channel assignment configurations that our SICA and GICA algorithms produce. The configurations produced by SICA and GICA are compared against two other *types* of configurations. Hence, we compare a total of four different types of configurations, which are explained as follows:

1. *Single-channel configuration*: All transmitter and receiver radios operate on the same channel and all flows entering and exiting a node are coupled onto the same radio of that node as long as the total magnitude of these flows is less than or equal to the maximum data rate of the radio. Each radio in the network is utilized at less than or equal to 1; if the total magnitude of a node's flows exceeds the maximum data rate, then the node's radios are maximally utilized in order, starting from radio 0.

2. *Random configuration*: The following steps generate a random flow-radio coupling and channel configuration for a given network:

   (a) Flows arriving at and departing from a node are coupled with the radios of the node in random with uniform distribution, taking care not to violate the feasibility constraint mentioned above (the total traffic bound on a radio should be less than or equal to the fastest data rate available).

   (b) Each link carrying traffic is assigned a random channel; however, links with common end points (radios) are assigned the same randomly selected channel.

3. *SICA configuration*: This is a flow-radio coupling and channel configuration computed by our SICA scheme.

4. *GICA configuration*: This is a flow-radio coupling and channel configuration computed by our GICA scheme.

A network topology is determined by the graph (N, F) and the set of the positions of the nodes, P. The nodes are placed in a rectangular grid and shortest-path multi-hop flows are generated between randomly selected source and destination node pairs. For the random assignment scheme, unless otherwise stated, for each network topology, a total of 100 random configurations are generated, and the average metrics over these 100 configurations are reported. Unless otherwise stated, for each simulation parameter set, 50 network topologies are generated. The reported metrics are the averages over these 50 topologies.

In Figures 6.2(a), 6.2(b), 6.2(c) and 6.2(d), we evaluate the performances of SICA and GICA as the network size increases with respect to the single-channel and random configuration schemes in terms of I_{ap}, I_{aph}, I_{awp} and R_{bc}, respectively. For all metrics, GICA, which considers network-wide interference yields better performance results than SICA, which in turn yields better results than both the single-channel and random configuration schemes (an exception to this occurs for the R_{bc} metric). However, for the I_{aph} metric, SICA performs nearly as well as GICA in spite of the fact that it considers node-local interference only. The reason is that I_{aph} considers the distance between interfering radios and SICA *tries to* put the radios of the same multi-radio node on non-overlapping channels as long as the multi-radio node does not have more active radios than the number of orthogonal channels available. And because such radios are the ones closest to each other, the performance gap between SICA and GICA decreases.

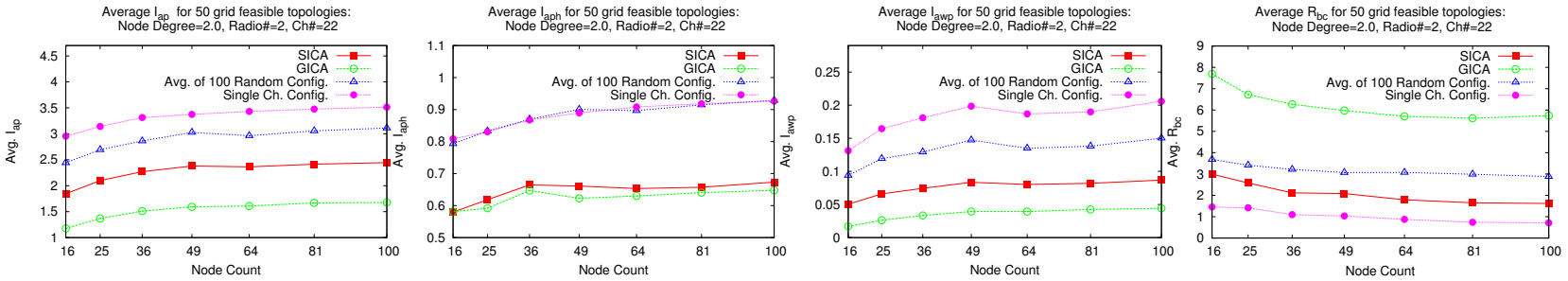
As Figure 6.2(d) reveals, GICA achieves 256% capacity improvement over SICA for a network of 16 nodes and about 354% improvement over SICA for 100 nodes. As the network size increases, the performance gap between SICA and GICA widens. GICA achieves up to 8 times capacity improvement over the single-channel configuration (for 100 nodes) and up to 208% improvement over random configuration (for 16 nodes). The random configuration scheme performs better than SICA in terms of R_{bc} for all network sizes. R_{bc} is modeled based on a protocol interference model, and SICA's local decisions lead to nodes choosing the same channels for their radios as their physical neighbors, which leads to increased interference and reduced network capacity. In this case, choosing the channels in random yields better results as Figure 6.2(d) reveals.

Figure 6.3 shows how I_{ap} changes as a function of the increasing number of wireless channels (M) for networks of 16, 36, 64 and 100 two-radio nodes. SICA does not make use of the increasing number of channels for any network size, whereas GICA effectively makes use of it and yields lower I_{ap} values for increasing M. Similar to SICA, the single-channel configuration has almost constant I_{ap} with respect to increasing M for a given network size, whereas, similar to GICA, the random configuration scheme can make use of the increasing number of channels to some extent. GICA can achieve an improvement of about 242% over the random configuration scheme for 16 nodes, and of about 215% for 100 nodes.

In terms of the I_{aph} metric, as Figure 6.4 reveals, SICA achieves better performance than GICA for 11 wireless channels for all network sizes under consideration. Because I_{aph} takes the distances between the radios into account, SICA achieves lower I_{aph} values than GICA for 11 channels by putting the co-located radios (of a node) on orthogonal channels. However, as the number of channels increases, GICA performs better than SICA, and as the network size increases, the performance gain of GICA over SICA increases (112% for $|N|= 16, M = 55$ and 117% for $|N|= 100, M = 55$).

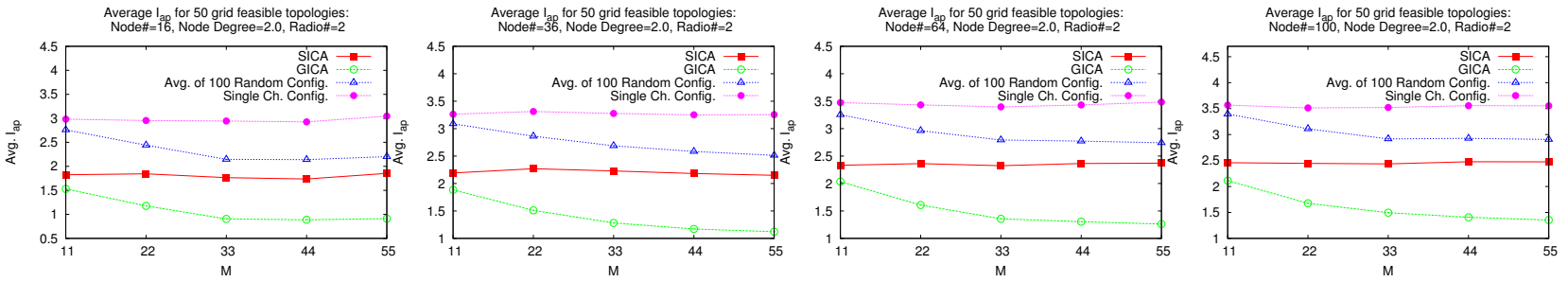
In Figure 6.5, we observe that the performance gains in terms of the I_{awp} metric achieved with GICA over SICA are even more pronounced when compared to the I_{aph} metric. GICA performs 519% better for $|N|= 16, M = 55$ and 306% better for $|N|= 100, M = 55$ than SICA in terms of I_{awp}.

Figure 6.6 shows the average R_{bc} of 50 topologies for networks of 16, 36, 64 and 100 nodes in relation with increasing M. GICA effectively uses the available channels and can increase the residual capacities as the number of available channels increases. As previously noted for Figure 6.2(d), the random configuration scheme achieves increased capacity when compared to SICA, however, GICA can achieve up to 207% improvement over the random scheme in terms of R_{bc} for a network of 16 nodes and with 55 wireless channels. And for a network of 100 nodes, the improvement achieved by GICA over the random scheme is over 200% with 55 channels.



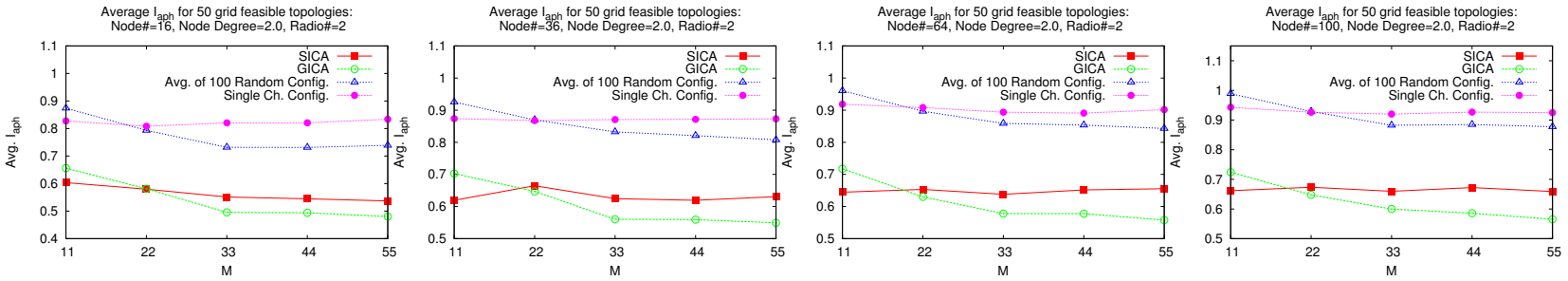
(a) I_{ap} (b) I_{aph} (c) I_{awp} (d) R_{bc}

Figure 6.2: Effects of the network size ($|N|$) on I_{ap}, I_{aph}, I_{awp}, and R_{bc}.



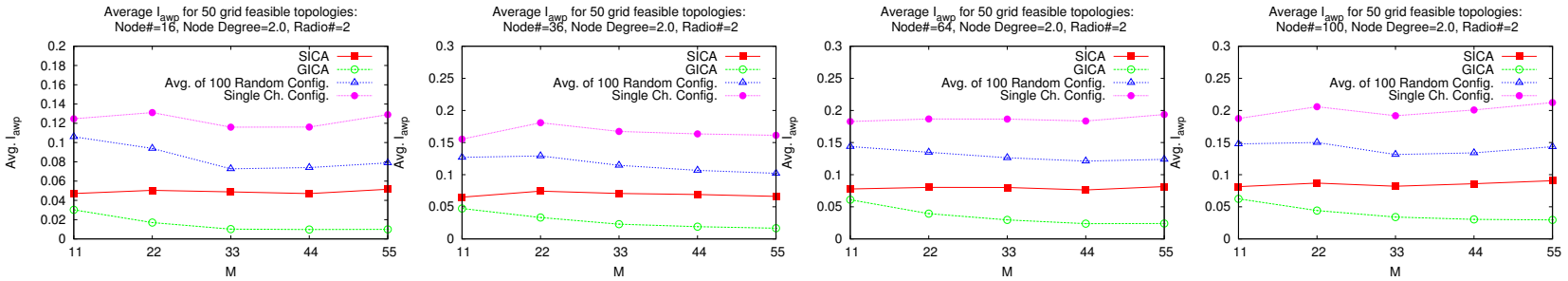
(a) $|N|= 16$ (b) $|N|= 36$ (c) $|N|= 64$ (d) $|N|= 100$

Figure 6.3: Effects of the number of available wireless channels (M) on I_{ap} for different network sizes.



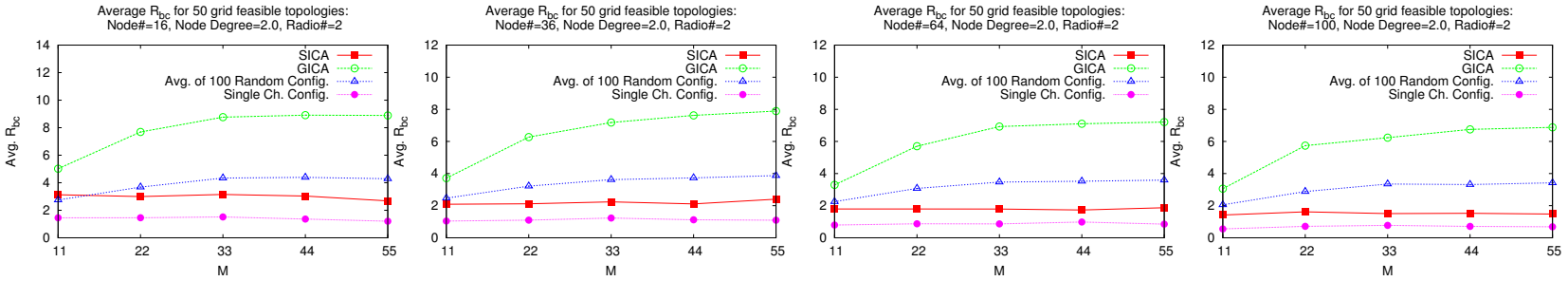
(a) $|N|= 16$ (b) $|N|= 36$ (c) $|N|= 64$ (d) $|N|= 100$

Figure 6.4: Effects of the number of available wireless channels (M) on I_{aph} for different network sizes.



(a) $|N|= 16$ (b) $|N|= 36$ (c) $|N|= 64$ (d) $|N|= 100$

Figure 6.5: Effects of the number of available wireless channels (M) on I_{awp} for different network sizes.



(a) $|N|= 16$ (b) $|N|= 36$ (c) $|N|= 64$ (d) $|N|= 100$

Figure 6.6: Effects of the number of available wireless channels (M) on R_{bc} for different network sizes.

6.4 Summary

Despite the importance of the flow-radio assignment problem in the context of multi-radio WMNs, few studies in the literature have attempted to address this problem in conjunction with the channel assignment problem. The heuristics and the centralized algorithms introduced in this chapter are amongst the first to jointly address these two problems.

The SICA scheme, which relies only on node-local information, lends itself straightforwardly to a distributed implementation. It also has low time and space complexity. The GICA scheme, however, makes use of global information and performs considerably better than SICA in terms of the metrics introduced in this chapter. With 22 wireless channels and 5 channels separation between orthogonal channels, GICA can achieve, over SICA, up to 157% improvement in terms of the average protocol interference metric and up to 297% improvement in terms of the average traffic-weighted protocol interference metric. When the random and single-channel configuration schemes are considered, the improvements achieved with GICA are even more pronounced: up to 207% and 251% respectively, in terms of the protocol interference metric, and up to 554% and 773% respectively, in terms of the traffic-weighted protocol interference metric.

Chapter 7

Distributed Joint Flow-Radio and Channel Assignment Using Partially Overlapping Channels in Multi-Radio Wireless Mesh Networks

In this chapter we present our *D*istributed *F*low-*R*adio *C*hannel *A*ssignment (DFRCA) solution, a distributed joint flow-radio and channel assignment scheme and the accompanying distributed protocol in the context of multi-channel multi-radio wireless mesh networks. The scheme's performance is evaluated on small networks for which the optimal flow-radio and channel configuration can be computed, as well as on large random topologies.

In Section 7.1, we discuss our main motivations in proposing a distributed protocol, list our contributions in this chapter, and define the problem scope for this chapter. Section 7.2 discusses our proposed distributed solution for the joint flow-radio and channel assignment problem. Using the performance metrics introduced in Section 6.3.1, Section 7.3 gives the simulation results obtained

for the proposed distributed scheme as well as for random and single-channel configurations. Section 7.4 concludes the chapter.

7.1 Introduction

As discussed in Section 2.3, as a consequence of the constraint that two radios must be tuned to the same channel for them to communicate with each other, flow-radio assignment, which determines the radio a flow to a neighboring node will use, has a direct impact on the performance achievable by the channel assignment. In the worst-case scenario, the channel assignment procedure may be obliged by the flow-radio assignment to use only one channel, making it ineffective. This scenario occurs when all the WMN's active (traffic-carrying, utilized) radios are connected in a single subgraph (as explained later in Section 7.2.2).

Despite the prominent impact of the flow-radio assignment on the performance of the channel assignment, few studies [41,42] in the literature have attempted to jointly address these two problems; to the best of our knowledge, our distributed joint flow-radio and channel assignment (DFRCA) scheme discussed here is the first to do so. The main contributions of our study are as follows:

- To the best of authors' knowledge, the joint handling of the flow-radio assignment and channel assignment problems within the framework of a distributed protocol is the first in the literature.
- Unlike most existing studies, we consider overlapping as well as orthogonal channels for channel assignment.
- We observe and take into account the WMN's traffic patterns, making the proposed scheme traffic load aware.
- The distributed scheme we propose is highly configurable and adaptable to different WMN topologies, to different wireless medium characteristics and to different wireless communication standards.

Symbol	Meaning
N	Multi-radio node set
F	One-hop flow set
n_i or i	Node i
P_i	Coordinates of n_i
α	Path loss exponent
d_T	Transmission range
d_I	Interference range
D	Radio interface count in a node
(i,k)	k^{th} radio interface of n_i
ρ_{max}	Maximum data rate of a radio
$f_{i,j,k,l,x}$	Flow from (i,k) to (j,l) on channel x
N_d	Avg. node degree for a random topology
M	Number of available wireless channels
O_Δ	Orthogonal channel separation
d_D	Delegation range

Table 7.1: Definitions of symbols and abbreviations.

This chapter shares the system model description and the underlying assumptions given in Section 5.2, with the exception that our aim in this chapter is to develop a distributed scheme to decide on flow-radio assignment and to compute the channels to be assigned to radios. The same set of symbols given in Table 6.1 is used in this chapter with the addition of d_D for the *delegation range* tunable parameter of our distributed scheme (see Section 7.2.3). Table 6.1 is replicated here, as Table 7.1, for ease of reference together with the addition of the new symbol.

7.2 A Distributed Scheme for Joint Flow-Radio and Channel Assignment

Our distributed joint flow-radio and channel assignment scheme consists of four phases, and each multi-radio node executes each phase in parallel. During the phases, a node shares information with its k-neighborhood, k being a parameter of our distributed scheme. k is chosen in relation to the interference range, d_I. A typical value for k is 2, which implies that a node initially exchanges messages only in its 2-neighborhood. Only at the final phase, where final channel selections are announced in the WMN, *might* a node have to exchange messages outside its

k-neighborhood to assure that radio links are actually established.

The scheme consists of the following four phases:

1. *F*low-*R*adio Assignment (FR) Phase
2. *T*ransmitter *A*nnouncement (TA) Phase
3. Channel *S*elector *E*lection (SE) Phase
4. *C*onflict *E*limination (CE) Phase

7.2.1 Flow-Radio Assignment Phase

In the FR-Phase, each node executes the concentrating flow-radio assignment heuristic given in Algorithm 6 (see Chapter 6). Since Algorithm 6 only uses information local to a node, it readily lends itself to a distributed implementation. At the end of the FR-Phase, a node has determined the flow-radio couplings in coordination with its neighbors.

7.2.2 Transmitter Announcement Phase

The TA-Phase collects information about all flow-radio assignments in a k-hop neighborhood. Because flow-radio assignment information is disseminated in the k-neighborhood of each node during this phase, at the end of it, each node has an estimate on the number of (single-hop) flows that can be decoupled from each other considering only its k-neighborhood.

In this context, decoupling flows means putting each flow in a k-neighborhood on a different channel, which mitigates inter-flow interference and, in the context of multi-hop flows, intra-flow interference. Of course, decoupling may not be feasible if there are not enough wireless channels and/or radios in a k-neighborhood. The channel configuration performed by such a *k-neighborhood local* algorithm may also fall far from a global optimum solution if the WMN's neighboring nodes

Algorithm 11 Transmitter Announcement Phase on n_i
Input: $C : F \rightarrow R$, flow-radio coupling information for n_i
Output: (N_k, F_k)
Output: Ψ, set of subgraphs in the k-neighborhood of n_i
Output: Ψ_c, colour classes of $G_c(\Psi, E)$

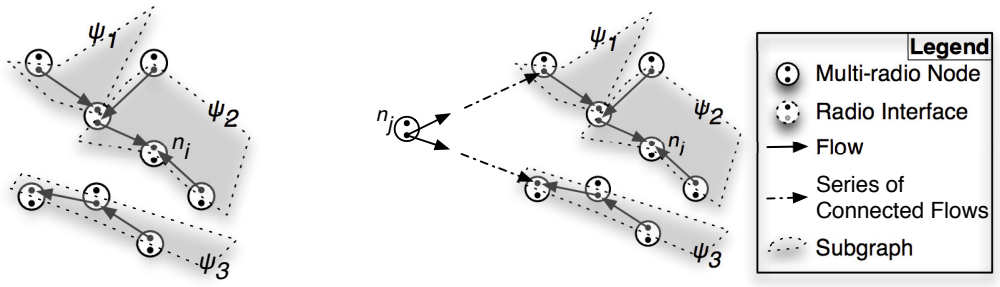
1: **procedure** PHASETA($C : F \rightarrow R$)
2: $H_1 \leftarrow \{j : |P_i - P_j| \leq d_T\}$ ▷ H_1 is the set of one-hop neighbors of n_i discovered via broadcasts
3: $H_k \leftarrow \varnothing$ ▷ H_k is the set of ids of the nodes in the k-neighborhood of n_i not in H_1
4: $TA_i \leftarrow (C, ttl = k)$ ▷ `TA` message of n_i
5: Broadcast TA_i
6: **for all** Unique TA_j received **do**
7: **if** $j \notin H_1$ **then**
8: $H_k \leftarrow H_k \cup j$
9: **end if**
10: $(N_k, F_k) \leftarrow (N_k, F_k) \cup TA_j.C$
11: $TA_j.ttl \leftarrow TA_j.ttl - 1$
12: **if** $TA_j.ttl > 0$ **then** ▷ Limited-scope flooding in k-neighborhood
13: Broadcast TA_j
14: **end if**
15: **end for**
16: $\Psi \leftarrow$ FINDSUBGRAPHS((N_k, F_k))
17: $G_c(\Psi, E) \leftarrow$ FINDCONFLICTGRAPH(Ψ)
18: $\Psi_c \leftarrow$ Vertex colouring classes of $G_c(\Psi, E)$
19: **end procedure**

do not *perceive* similar k-neighborhoods. However, for routing topologies where k-hop neighbors share similar k-neighborhoods, the TA-Phase, given in Algorithm 11, achieves intelligent channel assignment by correctly estimating the number of flows to be decoupled in the k-neighborhood.

A node starts the TA-Phase by exchanging flow-radio assignment information in its k-neighborhood. For this purpose, it broadcasts (on a common channel) a `TA` (Transmitter Announcement) message containing C, the flow-radio coupling information of itself, with a TTL set to k (see Algorithm 11). A node that receives an announcer node's `TA` message for the first time, decrements the TTL and broadcasts the message, unless the message's TTL is zero. As the node receives `TA` messages from its k-hop neighbors, it constructs its k-hop neighborhood set, H_k, and buffers the k-neighborhood flow-radio assignment information in (N_k, F_k).

After the `TA` messages have been exchanged, the node proceeds to calculate the set of disjoint k-neighborhood *subgraphs*, Ψ. The term subgraph defines a set

of radios (vertices) connected with incident flows (edges). Two disjoint subgraphs in a node's k-neighborhood share no common radios (see Figure 7.1(a)). Hence, if there are enough physical channels, each subgraph may operate on a distinct, possibly non-overlapping channel. Outside the node's k-neighborhood these two subgraphs may be connected, in which case they will have to operate on the same channel (see Figure 7.1(b)).



(a) n_i perceives that it may be possible to operate ψ_1, ψ_2 and ψ_3 on distinct (possibly non-overlapping) channels.

(b) However, two or more subgraphs may in reality be connected outside n_i's k-neighborhood.

Figure 7.1: k-neighborhood subgraphs, $\Psi = \{\psi_1, \psi_2, \psi_3\}$, of n_i.

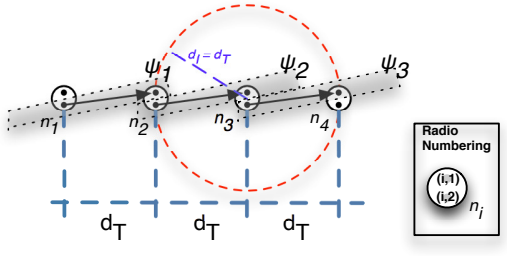
Ψ is computed using Algorithm 12, which leaves the flows whose transmitters are not in n_i's k-neighborhood out of Ψ. This procedure is motivated by the effort to reuse channels *outside* the k-neighborhood of a node under consideration.

After computing Ψ, the node constructs the conflict graph of Ψ, $G_c(\Psi, E)$, using Algorithm 13. An edge (ψ_1, ψ_2) is added to $G_c(\Psi, E)$ whenever a transmitter radio in ψ_1 interferes with a receiver radio in ψ_2 (see Figure 7.2). After computing $G_c(\Psi, E)$, the node then calls a greedy vertex colouring heuristic to find the set of colour classes [76], Ψ_c, of $G_c(\Psi, E)$. $|\Psi_c|$, which approximates the chromatic number [76] of G_c, $(\chi(G_c))$, is an upper bound on the minimum number of channels needed for all the subgraphs in Ψ to decouple. Considering $\chi(G_c)$ instead of $\chi(\Psi)$ promotes the spatial reuse of the channels *inside* the k-neighborhood.

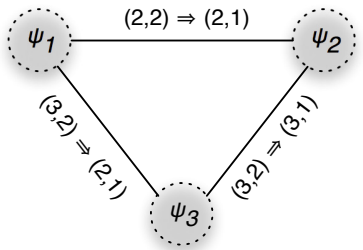
By the end of the TA-Phase, the set of one-hop neighbors, H_1, and the set of k-hop neighbors, H_k, are available for the remaining phases of the distributed scheme.

Algorithm 12 Computation of the Set of k-neighborhood Subgraphs on n_i
Input: (N_k, F_k)
Output: Ψ, the set of subgraphs

```
1:  procedure FindSubgraphs((N_k, F_k))
2:    Ψ ← ∅                          ▷ Ψ is the set of subgraphs. The number of subgraphs will be |Ψ|
3:    for all f_{i',j',k',l'} ∈ F_k do
4:      if i' ∉ H_1 ∧ i' ∉ H_k ∧ i' ≠ i then
5:        continue
6:      end if
7:      R_t ← (i', k')
8:      R_r ← (j', l')
9:      R_o ← (−1, −1)
10:     for all ψ_1 ∈ Ψ do
11:       if R_t ∈ ψ_1 then
12:         R_o ← R_r
13:       else if R_r ∈ ψ_1 then
14:         R_o ← R_t
15:       end if
16:       if R_o ≠ (−1, −1) then
17:         ψ_1 ← ψ_1 ∪ {R_t, R_r}
18:         for all ψ_2 ∈ Ψ do
19:           if ψ_2 ≠ ψ_1 ∧ R_o ∈ ψ_2 then
20:             ψ_1 ← ψ_1 ∪ ψ_2
21:             Ψ ← Ψ \ {ψ_2}
22:           end if
23:         end for
24:       end if
25:     end for
26:     if R_o = (−1, −1) then        ▷ Then no other subgraph contains f_{i',j',k',l'}
27:       ψ_new ← {R_t, R_r}
28:       Ψ ← Ψ ∪ {ψ_new}
29:     end if
30:   end for
31: end procedure
```



(a) Ψ of a linear topology, where $d_I = d_T$.



(b) $G_c(\Psi, E)$ for Figure 7.2(a). $(i,k) \implies (j,l)$ denotes that transmitter (i,k) interferes with receiver (j,l).

Figure 7.2: k-neighborhood subgraphs of n_i.

Algorithm 13 Computation of the k-neighborhood Conflict Graph on n_i
Input: Ψ
Output: $G_c(\Psi, E)$, the conflict graph of the k-neighborhood subgraphs

```
1:  procedure FindConflictGraph(Ψ)
2:      E ← ∅                                              ▷ The edge set of G_c
3:      for all ψ_1 ∈ Ψ do
4:          for all (i, k) ∈ ψ_1 do
5:              for all f_{i,j,k,l} ∈ F do
6:                  for all ψ_2 ∈ (Ψ \ {ψ_1}) do
7:                      for all (j', l') ∈ ψ_2 do
8:                          if (ψ_1, ψ_2) ∈ E then
9:                              break
10:                         end if
11:                         for all f_{i',j',k',l'} ∈ F do
12:                             if d(P_i, P_{j'}) ≤ d_I then
13:                                 E ← E ∪ {(ψ_1, ψ_2)}
14:                                 break
15:                             end if
16:                         end for
17:                     end for
18:                 end for
19:             end for
20:         end for
21:     end for
22: end procedure
```

7.2.3 Channel Selector Election Phase

In this phase, a node determines the subgraphs of its k-neighborhood (a subset of Ψ) for which it will select channels, and becomes the *manager* of those subgraphs. The node also estimates the managers of the remaining subgraphs in Ψ. Such nodes are called *remote* managers with respect to the node under discussion. Having estimated the number of distinct channels needed in its k-neighborhood, the node then proceeds to determine those channels.

The SE-Phase begins by sending and receiving unicast `SE` (Selector Election) messages in the k-neighborhood. Algorithm 14 outlines this phase. Each node tells its k-hop neighbors the subgraph count in its k-neighborhood, $|\Psi|$, and its set of colour classes, Ψ_c. The node builds two tables using the `SE` messages it receives. The first table, $M_{|\Psi|}$, holds the subgraph counts of the nodes in the k-neighborhood, and the second table, M_{Ψ_c}, holds their sets of colour classes. After these two tables are built, the node iterates over all the radios in each of its

Algorithm 14 Channel Selector Election Phase on n_i
Input: H_1, set of one-hop neighbors of n_i
Input: H_k, set of k-hop neighbors of n_i not in H_1
Input: Ψ, set of k-neighborhood subgraphs
Input: Ψ_c, colour classes of Ψ
Output: S_i, set of sets of radios on Ψ whose channels are to be selected by n_i

1: **for all** $j \in (H_1 \cup H_k)$ **do**
2: $SE_i \leftarrow (|\Psi|, \Psi_c)$ ▷ `SE` message of n_i
3: Send SE_i to n_j
4: **end for**
5: $S_i \leftarrow \varnothing$ ▷ set of colour classes of locally managed radios
6: $S_R \leftarrow \varnothing$ ▷ set of colour classes of remotely managed radios
7: $T \leftarrow \varnothing$ ▷ set of remotely managed radios
8: $M_{|\Psi|}[i] \leftarrow |\Psi|$ ▷ node id, $|\Psi|$ mappings
9: $M_{\Psi_c}[i] \leftarrow \Psi_c$ ▷ node id, Ψ_c mappings
10: **for all** SE_j received **do**
11: $M_{|\Psi|}[\mathrm{j}] \leftarrow SE_j.|\Psi|$
12: $M_{\Psi_c}[\mathrm{j}] \leftarrow SE_j.\Psi_c$
13: **end for**
14: **for all** $\psi \in \Psi_c$ **do**
15: **for all** $(i,k) \in \psi$ **do**
16: **if** $(i,k) \in T$ **then**
17: `continue`
18: **end if**
19: $m \leftarrow$ SelectorId$((i,k), M_{|\Psi|}, M_{\Psi_c})$
20: $\psi_r \leftarrow \{(i',k') : \exists\, \psi'_r \in M_{\Psi_c}[m], (i,k) \in \psi'_r \wedge (i',k') \in \psi'_r\}$
21: **if** $\psi_r \notin (S_R \cup S_i)$ **then**
22: $M_I[\psi_r] \leftarrow m$
23: **if** $m \neq i$ **then** ▷ Then a remote node manages the radio
24: $S_R \leftarrow S_R \cup \{\psi_r\}$
25: $M_C[m] \leftarrow M_C[m] + 1$
26: **else**
27: $S_i \leftarrow S_i \cup \{\psi_r\}$
28: **end if**
29: **end if**
30: **if** $m \neq i$ **then**
31: $T \leftarrow T \cup \{(i',k') : \exists\, \psi' \in \Psi, (i,k) \in \psi' \wedge (i',k') \in \psi'\}$ ▷ add all radios on the subgraph of (i,k) to T
32: **end if**
33: **end for**
34: **end for**
35: PrepareDlgMap(Ψ, S_i, T)
36: DoChAllotment(S_i, S_R, M_I, M_C)

colour classes to determine the manager that will select a channel for each radio, using Algorithm 15.

Algorithm 15 Select Radio Manager
Input: (i, k), radio whose channel selector is to be determined
Input: $M_{|\Psi|}$
Input: M_{Ψ_c}
Output: m, channel selector node's id

```
1:  procedure SelectorId((i,k), M_|Ψ|, M_Ψc)
2:      m ← −1
3:      c_max ← 0
4:      d_min ← ∞
5:      for all i′ ∈ keys[M_Ψc] do
6:          for all ψ ∈ M_Ψc[i′] do
7:              if (i,k) ∈ ψ ∧ (m = −1 ∨ M_|Ψ|[i′] > c_max ∨ (M_|Ψ|[i′] = c_max ∧ |P_i′ − P_i| >
    d_D ∧ d_min > |P_i′ − P_i|) ∨ (M_|Ψ|[i′] = c_max ∧ (|P_i′ − P_i| ≤ d_D ∨ |P_i′ − P_i| = d_min) ∧ m > i′))
    then
8:                  m ← i′
9:                  c_max ← |M_|Ψ|[i′]|
10:                 d_min ← |P_i′ − P_i|
11:             end if
12:         end for
13:     end for
14: end procedure
```

In Algorithm 15, the node that contains the radio in one of its colour classes and has the highest subgraph count (highest $|\Psi|$) is selected as the manager. Nodes with higher subgraph counts are given priority for selecting channels because they can decouple more subgraphs. Of the nodes with equal subgraph counts, outside the delegation range (explained later in this section) spatially closer nodes are preferred. Inside the delegation range nodes with smaller ids are preferred. Because radios in a subgraph must operate on the same channel, once the manager of a radio is determined, all other radios in the same subgraph are assigned the same manager.

As the node determines the managers of the radios in its k-neighborhood, it builds a set of remotely managed colour classes, S_R, and a set of the colour classes it manages, S_i. It notes the selected manager of each colour class in the table M_I.

As will be explained later in this section, a manager uses its colour classes to select channels for the radios it manages. For channels selected by different managers of the same k-neighborhood to be as spectrally far as possible from

each other, a mechanism for coordinating the colour classes of these *spatially close* managers is needed. For this purpose, we define the delegation range, d_D. Inside a circular region of radius d_D in a node's k-neighborhood, the colour classes, hence the channel selections, of managers are coordinated. This coordination ceases outside d_D.

In Figure 7.3, we give an example for the coordination need that may arise between managers. The node m announces to m' its set of colour classes, $\Psi_c = \{\psi_{c1}, \psi_{c2}\}$. m' determines its own colour classes, $\Psi'_c = \{\psi'_{c1}\}$, which implies that it is responsible for selecting a channel for the radios in ψ'_{c1}. However, m' realizes that $\psi'_{c1} \cap \psi_{c1} \neq \varnothing$, and delegates the management of the radios in $\psi'_{c1} \setminus \psi_{c1}$ to manager m because m is inside the delegation range. Algorithm 16 (which outlines these steps) stores the delegation mappings in M_D to be used during the CE-Phase.

Increasing d_D decreases the parallelism achieved by the distributed channel assignment procedure. However, especially for long chain topologies, increasing d_D also substantially decreases the intra-flow interference in the network (the effects of d_D on such interference are explored in Section 7.3).

By the end of the SE-Phase, in S_i and S_R the node contains an estimation of its k-neighborhood channel selectors (which k-hop neighbors will select channels for which sets of radios). The node can now intelligently assign channels to the subgraphs it is responsible for (S_i) by efficiently using the channel space available in its k-neighborhood. The channel allotment heuristic is given in Algorithm 17.

Algorithm 17 starts by building the weighted conflict graph, $G_c(S_A, E)$, of $S_A = S_R \cup S_i$. $G_c(S_A, E)$ is later used in the SE-Phase for intelligently mapping selected channels to the colour classes of the k-neighborhood. The computation of $G_c(S_A, E)$ is similar to that of $G_c(\Psi, E)$ and is given in Algorithm 18. The weight of the undirected edge (ψ_{c1}, ψ_{c2}) in G_c estimates the total physical interference between the colour classes ψ_{c1} and ψ_{c2} assuming both colour classes operate on the same wireless channel and is calculated as follows:

Algorithm 16 Prepare Delegation Map
Inputs: Ψ, S_i, T
Output: M_D, the dictionary that holds the master radio of a remotely managed slave radio

```
1: procedure PrepareDlgMap(Ψ, S_i, T)
2:   for all ψ_c ∈ S_i do
3:     ψ_d ← ψ_c \ T
4:     if ψ_d ≠ ψ_c then                ▷ if this colour class is remotely managed by radio (m, k_m)
5:       for all (i, k) ∈ ψ_d do
6:         ψ ← {(i', k') : ∃ ψ' ∈ Ψ, (i, k) ∈ ψ' ∧ (i', k') ∈ ψ'}      ▷ ψ is the subgraph of radio (i, k)
7:         d_min ← ∞
8:         (i_d, k_d) ← (−1, −1)
9:         for all (i', k') ∈ ψ do
10:          if |P_i' − P_m| ≤ d_min then
11:            (i_d, k_d) ← (i', k')
12:            d_min ← |P_i' − P_m|
13:          end if
14:        end for
15:        if d_min ≤ d_D then                ▷ d_D is the delegation range
16:          M_D[(i_d, k_d)] ← (m, k_m)
17:          ψ_d ← ψ_d \ ψ
18:        end if
19:      end for
20:      if ψ_d = ∅ then
21:        S_i ← S_i \ {ψ_c}
22:      else
23:        ψ_c ← ψ_d
24:      end if
25:    end if
26:  end for
27: end procedure
```

Algorithm 17 Channel Allotment Algorithm Running on n_i
Input: S_i, set of sets of radios n_i is responsible for selecting channels
Input: S_R, set of colour classes of remotely managed radios
Input: M_I, dictionary of manager node ids for the colour classes in S_i and S_R
Input: M_C, dictionary that holds the number of channels a k-hop neighbor is expected to select
Output: L_S, list of $|S_i|$ channels selected, one for each of the sets of radios in S_i

```
1: procedure DoChAllotment(S_i, S_R, M_I, M_C)
2:   S_A ← S_R ∪ S_i
3:   G_c(S_A, E), W_E ← FindWeightedConflictGraph(S_A)
4:   L ← ChList(|S_A|)
5:   M_C[i] ← |S_i|
6:   M_L ← ChSelection(L, M_C, −1, |S_i|)
7:   ChDist(S_i, G_c(S_A, E), W_E, M_I, M_L)
8: end procedure
```

$$W_{(j',l')}(\psi_c) = \sum_{\substack{f_{i,j,k,l} : (i,k) \in \psi_c \wedge \\ |P_i - P_{j'}| \leq d_I}} \frac{1}{|P_i - P_{j'}|^{\alpha}}$$

$$W(\psi_{c2}, \psi_{c1}) = \sum_{f_{i',j',k',l'} : (j',l') \in \psi_{c2}} W_{(j',l')}(\psi_{c1}) \tag{7.1}$$

$$W_E[(\psi_{c1}, \psi_{c2})] = W(\psi_{c1}, \psi_{c2}) + W(\psi_{c2}, \psi_{c1}),$$

where $W(\psi_{c2}, \psi_{c1})$ is an estimation of the total physical interference caused by all transmitters in ψ_{c1} on each receiver of ψ_{c2}. The edge weights of G_c are stored in the dictionary W_E by Algorithm 18.

Algorithm 18 Computation of the Colour Classes' Weighted Conflict Graph on n_i
Input: S_A
Output: $G_c(S_A, E)$, conflict graph of the colour classes
Output: W_E, dictionary of edge weights of $G_c(S_A, E)$

```
1:  procedure FindWeightedConflictGraph(S_A)
2:      E ← ∅                                              ▷ The edge set of G_c
3:      for all ψ_1 ∈ S_A do
4:          for all (i,k) ∈ ψ_1 do
5:              for all f_{i,j,k,l} ∈ F do
6:                  for all ψ_2 ∈ (S_A \ {ψ_1}) do
7:                      for all (j',l') ∈ ψ_2 do
8:                          if |P_i − P_{j'}| > d_I then
9:                              continue
10:                         end if
11:                         for all f_{i',j',k',l'} ∈ F do
12:                             E ← E ∪ {(ψ_1, ψ_2)}
13:                             if |P_i − P_{j'}| < 1.0 then
14:                                 d ← 1.0
15:                             else
16:                                 d ← |P_i − P_{j'}|
17:                             end if
18:                             W_E[(ψ_1,ψ_2)] ← W_E[(ψ_1,ψ_2)] + 1/d^α     ▷ α is the path loss exponent
19:                         end for
20:                     end for
21:                 end for
22:             end for
23:         end for
24:     end for
25: end procedure
```

Algorithm 17 then prepares a list of channels, L, to be used in the k-neighborhood by calling Algorithm 19. To minimize interference between subgraphs (grouped as colour classes) in the k-neighborhood, Algorithm 19 fills L with $|S_A|$ channels as spectrally far as possible from each other. After L is filled, Algorithm 20 is called to prepare a dictionary of channel lists, M_L, which maps manager ids in the k-neighborhood to the estimated channel selection lists. The list of channels to be used for colouring S_i is then given by $M_L[i]$. To determine the $|S_i|$ channels to be used out of L, Algorithm 20 employs the heuristic given in Algorithm 22, whose main motivation is to select $|S_i|$ channels as spectrally far as possible from each other. For example, if $L = [1, 6, 11]$ and two channels are to be selected ($|S_i|= 2$), the heuristic selects channels 1 and 11. Or if $L = [5, 6, 7]$, the heuristic selects channels 5 and 7.

Algorithm 19 Channel List Initialization Algorithm Running on n_i
Input: $|S_A|$, cardinality of the set of colour classes of remotely and locally managed radios
Output: L, channel list

```
1:  procedure ChList(|S_A|)
2:      δ ← 0
3:      f ← 1
4:      if |S_A|= 1 then                          ▷ Then randomly select a channel
5:          f ← A random channel
6:      else
7:          δ ← (M−1)/(|S_A|−1)                   ▷ M is the number of available wireless channels
8:      end if
9:      for all i ∈ ℤ ∧ i ∈ [0, |S_A|) do
10:         ch ← f + i δ
11:         Round ch to the nearest integer
12:         if ch > M then
13:             ch ← M
14:         end if
15:         L_i ← ch
16:     end for
17: end procedure
```

At the end of the channel allotment, as the final step of the SE-Phase, the channels in L are distributed to the colour classes in $S_A = S_R \cup S_i$ using $G_c(S_A, E)$ with the heuristic given in Algorithm 23. For traversing G_c, vertex weights, W_V, are calculated. The weight of a vertex is the sum of the incident edge weights. A vertex with a higher weight implies a colour class (a set of subgraphs) that puts/receives higher levels of interference on/from the other colour classes that are incident to it in G_c than a vertex with a lower weight. Hence, vertices with

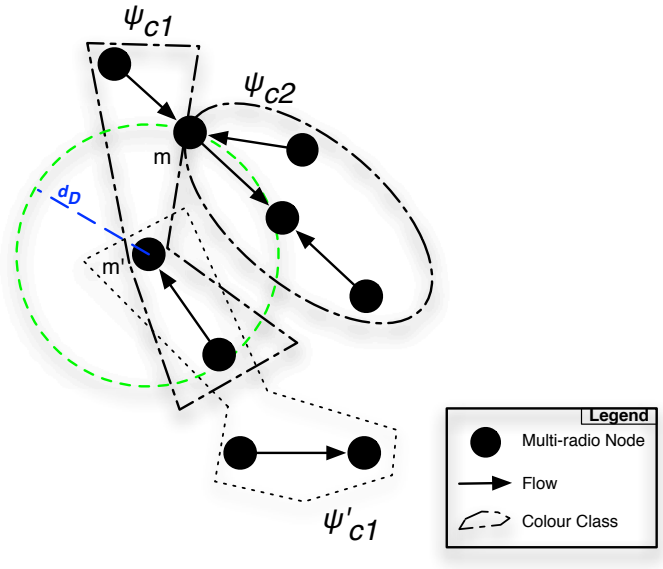


Figure 7.3: Coordination need for colour classes of k-neighbor manager nodes from the point of view of m'.

Algorithm 20 Channel Selection Estimation for the k-neighborhood of n_i
Input: L, list of available (not yet assigned, free) channels
Input: M_C
Input: m, current channel selector node's id
Input: c, number of channels to be selected for n_m
Output: M_L, node id-selected channels list mappings

```
1:  procedure ChSelection(L, M_C, m, c)
2:      if c = 0 then
3:          return
4:      end if
5:      l ← LeastPriorNode(M_C)
6:      if l ≠ −1 then
7:          c' ← M_C[l]
8:          Del M_C[l]                          ▷ Remove key l from M_C
9:          ChSelection(L, M_C, l, c')
10:     end if
11:     if m = −1 then
12:         return
13:     end if
14:     L_S ← SelectCh(L, c)
15:     M_L[m] ← L_S
16: end procedure
```

Algorithm 21 Least Prior Node Selection on n_i
Input: M_C
Output: l, least prior node's id

```
1: procedure LeastPriorNode(M_C)
2:     c_min ← ∞
3:     l ← −1
4:     d_max ← −1
5:     for all j ∈ keys[M_C] do
6:         if c_min > M_C[j] ∨ (c_min = M_C[j] ∧ |P_i − P_j| > d_D ∧ d_max < |P_i − P_j|) ∨ (c_min =
       M_C[j] ∧ (|P_i − P_j| ≤ d_D ∨ d_max = |P_i − P_j|) ∧ l < j) then
7:             l ← j
8:             c_min ← M_C[j]
9:             d_max ← |P_i − P_j|
10:        end if
11:    end for
12: end procedure
```

higher weights are given priority over vertices with lower weights during traversal. Breadth-first traversal of the graph starts with the heaviest vertex (see Algorithm 24). Next, the incident vertices of the currently visited vertex are visited. As each vertex is visited, the channel minimizing the total interference between the *previously* visited vertices and the current vertex is assigned to the vertex (see Algorithm 23). This minimum-interference-channel is selected according to the cost function given in (7.2) for the current vertex v:

$$I_c(v) = \sum_{w : w \in S_A \wedge M_V[w] \neq -1} \frac{W_E[(v,w)]}{|c - M_V[w]|}, \tag{7.2}$$

where c is a channel in L, M_V is the dictionary that holds the colour class-channel mappings (see Algorithm 23) and W_E is the edge weights of G_c. $M_V[w] = -1$ indicates that w has not been visited yet.

This scheme ensures that heavily interfering subgraphs are given priority for channel assignment and are assigned channels as spectrally far as possible from each other.

Algorithm 22 Channel Selection from Available Channels
Input: L, list of available (not yet assigned, free) channels
Input: c, number of channels to be selected
Output: L_S, list of c channels selected from L

```
1:  procedure SelectCh(L, c)
2:      δ ← 0
3:      L_S ← ∅
4:      if c ≤ 1 then                ▷ Then the median of the available channels list is selected
5:          Append median[L] to L_S
6:          Remove one instance of median[L] from L
7:      else
8:          δ ← (M−1)/(c−1)          ▷ M is the number of available wireless channels
9:      end if
10:     for all i ∈ ℤ ∧ i ∈ [0, c) do
11:         ch ← 1 + i δ
12:         Round ch to the nearest integer
13:         if ch > M then
14:             ch ← M
15:         end if
16:         if ch ∈ L then
17:             selectedCh ← ch
18:         else                     ▷ Then select the closest channel to ch from L
19:             dist ← ∞
20:             for all channel ∈ L do
21:                 if |ch − channel| < dist then
22:                     selectedCh ← channel
23:                     dist ← |ch − channel|
24:                 end if
25:             end for
26:         end if
27:         Append selectedCh to L_S
28:         Remove selectedCh from L
29:     end for
30: end procedure
```

Algorithm 23 Distribute Channels Using Colour Classes' Conflict Graph on n_i
Input: S_i, set of sets of radios n_i is responsible for selecting channels
Input: $G_c(S_A, E)$, colour classes' conflict graph
Input: W_E, dictionary of edge weights of $G_c(S_A, E)$
Input: M_I, dictionary of manager node id's for the colour classes in S_A
Input: M_L, node id-selected channels list mappings
Output: π_i, candidate channel configurations for the radios of n_i
Output: M_R, dictionary of remotely managed radios' channels

1: **procedure** ChDist(S_i, $G_c(S_A, E)$, W_E, M_I, M_L)
2: **if** $|S_i| = 0$ **then**
3: `return`
4: **end if**
5: $M_V \leftarrow \varnothing$ ▷ initialize a dictionary that holds vertex-channel mappings
6: $v \leftarrow$ TraverseNext(**nil**, $G_c(S_A, E)$, W_E)
7: **while** $v \neq$ **nil do**
8: $\forall c, I_c \leftarrow 0.0$
9: **for all** $c \in M_L[M_I[v]]$ **do** ▷ for each candidate channel c
10: **for all** $(v, w) \in E$ **do**
11: **if** $w \in$ keys$[M_V]$ **then** ▷ a channel has been selected for w
12: $I_c \leftarrow I_c + \frac{W_E[(v,w)]}{|c - M_V[w]|}$
13: **end if**
14: **end for**
15: **end for**
16: $M_V[v] \leftarrow c$, such that I_c is minimum in I ▷ channel c is selected for colour class v
17: $M_L[M_I[v]] \leftarrow M_L[M_I[v]] \setminus M_V[v]$ ▷ Remove channel c from candidate channels for node $M_I[v]$
18: $v \leftarrow$ TraverseNext(v, $G_c(S_A, E)$, W_E)
19: **end while**
20: **for all** $v \in S_i$ **do**
21: **for all** $(i', k') \in v$ **do**
22: **if** $i' = i$ **then**
23: $\pi_{i'}^{k'} \leftarrow M_V[v]$
24: **else**
25: $M_R[(i', k')] \leftarrow M_V[v]$
26: **end if**
27: **end for**
28: **end for**
29: **end procedure**

Algorithm 24 Traverse Next Vertex
Input: v, current vertex being visited
Input: $G_c(S_A, E)$
Input: W_E
Output: v, next vertex to be visited

```
1: procedure TraverseNext(v, G_c(S_A, E), W_E)
2:    ∀v ∈ S_A, W_V[v] ← Σ_{(v,w)∈E} W_E[(v, w)]   ▷ Vertex weight is the sum of incident edges' weights
3:    if v ≠ nil then
4:       Select (v, w) ∈ E such that W_E[(v, w)] is maximum in W_E and w has not been visited yet
5:       if w ≠ nil then                                   ▷ if such a w exists
6:          return w
7:       end if
8:    end if
9:    Select v ∈ S_A such that W_V[v] is maximum in W_V and v has not been visited yet
10:   return v                                   ▷ v is nil if no such vertex exists
11: end procedure
```

7.2.4 Conflict Elimination Phase

After the SE-Phase completes, manager nodes will have determined *candidate* channels for the radios they are responsible for. However, radios connected with a path in (N, F) may have been assigned different channels if the nodes responsible for assigning channels are neighbors of greater than k hops in (N, F), and if those nodes have selected conflicting channels for the radios. If these radios are actually assigned different channels, then the physical links that should exist between them will break. We call this situation a *conflict*, and the goal of this CE-Phase is twofold:

1. Eliminating any conflicts that may have arisen in the SE-Phase.
2. Announcing the selected channels to the other neighbors that have delegated this task to the manager nodes.

During the CE-Phase, the selected channel information will be negotiated and any conflicts will be resolved. Algorithm 25 outlines the CE-Phase, and it tries to determine the channel selected by the node with the largest number of subgraphs in its k-neighborhood and is the most heavily loaded node with the smallest node

id. Layer 3 or layer 2 addresses can be employed as the node ids and we assume that the employed ids are unique throughout the network.

Algorithm 25 Conflict Elimination Phase on n_i
Input: S_i, the set of sets of radios n_i is responsible for selecting channels
Input: π_i, candidate channel configurations for the radios in S_i
Input: M_R, the dictionary of remotely managed radios' channels
Input: M_D, the dictionary that holds the master radio of a remotely managed slave radio

1: $\Pi_k \leftarrow \varnothing,\ \forall k \in [1, D]$ ▷ Π_k is the set of channel selection announcements, $\left(|\Psi_j|, X_j, j, k', \pi_j^{k'}\right)$, received for radio (i, k). $|\Psi_j|$ is the number of subgraphs in the k-hop neighborhood of n_j. X_j is the magnitude of the total inbound/outbound traffic on n_j
2: $M_P[(i, k)] \leftarrow \varnothing,\ \forall k \in [1, D]$ ▷ Initially empty proxy tables
3: AnnounceSelections(S_i, M_D, π, Π, C)
4: **while** *true* **do**
5: Receive `CS` message $\left(|\Psi_j|, X_j, j, k', \pi_j^{k'}\right)$ or `DR` message $((j, k'))$ for (i, k)
6: **if** `DR` message received **then**
7: $M_P[(i, k)] \leftarrow M_P[(i, k)] \cup \{(j, k')\}$
8: **for all** $(z, l) \in M_P[(i, k)]$ **do** ▷ Announce to delegated radios
9: Send `CS` message, $(\Psi_{max_k}, X_{max_k}, N_{min_k}, D_{min_k}, C_k)$, to (z, l)
10: **end for**
11: **else** ▷ then `CS` message received
12: HandleCSAnnouncement(Π, M_P)
13: **end if**
14: **end while**
15: **for** $k = 1$ **to** D **do**
16: $\pi_i^k \leftarrow C_k$
17: **end for**

A node starts the CE-Phase by announcing the channel selections of the radios for which it is a manager by sending unicast `CS` (Channel Selection Announcement) messages. A `CS` message contains the selected wireless channel, the subgraph count ($|\Psi|$) of the origin node, the magnitude of the total inbound/outbound traffic on the origin node (X) and the node's unique id (see Algorithm 26). If the node is a manager for one of its own radios, then it sends the associated `CS` message to all one-hop neighbors of that radio on the radio's subgraph. If the node is a manager of a remote radio that is not connected to any of the node's own radios in the node's k-neighborhood, then the node sends the associated `CS` message in a multi-hop manner to the owner of the remote radio.

As the node receives a `CS` announcement, it determines the manager of the associated radio by selecting the node among the announcers with the highest subgraph count and the highest traffic but with the smallest id (see Algorithm 27). If

Algorithm 26 Announce Channel Selections on n_i
Input: S_i, set of sets of radios n_i is responsible for selecting channels
Input: M_D, dictionary that holds the master radio of a remotely managed slave radio
Input: π, candidate channel configurations for the radios in S_i
Input: Π, list of sets of channel selection announcements
Input: C, list of channel selection variables

1: **procedure** AnnounceSelections(S_i, M_D, π, Π, C)
2: **for all** $\psi \in S_i$ **do**
3: **for all** $(i', k') \in \psi$ **do**
4: **if** $i' = i$ **then**
5: $C_{k'} \leftarrow \pi_{i'}^{k'}$ ▷ Initialize channel selection variables
6: $\Pi_{k'} \leftarrow \Pi_{k'} \cup \left\{ (|\Psi_i|, X_i, i, k', \pi_{i'}^{k'}) \right\}$
7: Send CS message, $\left(|\Psi_i|, X_i, i, k', \pi_{i'}^{k'}\right)$, to one-hop neighbors on the subgraph of (i', k')
8: **else**
9: Send CS message, $(|\Psi_i|, X_i, i, k', M_R[(i', k')])$, to (i', k')
10: **end if**
11: **end for**
12: **end for**
13: **for all** $(i', k') \in \text{keys}[M_D]$ **do** ▷ Send delegation requests
14: Send DR message, $((i', k'))$, to $M_D[(i', k')]$
15: **end for**
16: **end procedure**

a new CS announcement changes the previously selected manager for a radio, then the node receiving the announcement announces the new selection to all one-hop neighbors on that radio's subgraph. Otherwise, the receiver of the announcement makes no new announcements.

7.3 Validation and Evaluation

To validate our distributed scheme and evaluate its performance, we simulate it in a custom environment based on the *CSIM for Java* [77] simulation engine, which is a library for developing discrete-event simulations. We develop a packet-based simulator that can truly simulate our distributed scheme using message exchanges among nodes simulating our multi-radio routers.

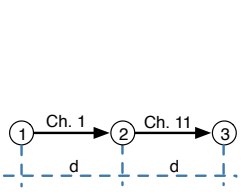
Next, we first describe how we validate our scheme using small topologies, for which it is easy to compute the optimal configurations. Then we present

Algorithm 27 Handle CS Announcement on n_i
Input: Π, list of sets of channel selection announcements
Input: M_P, proxy dictionary
Input: k, radio id for which the announcement has been received

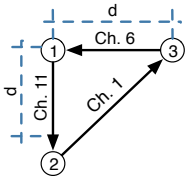
1: **procedure** HANDLECSANNOUNCEMENT(Π, M_P)
2: $$\Psi_{max_k} \leftarrow \max\left\{|\Psi_z|\text{: for } \exists(X_z, z, l, \pi_z^l),\ (|\Psi_z|, X_z, z, l, \pi_z^l) \in \Pi_k\right\}$$
3: $$X_{max_k} \leftarrow \max\left\{X_z : \text{for } \exists\left(\Psi_{max_k}, z, l, \pi_z^l\right),\ (\Psi_{max_k}, X_z, z, l, \pi_z^l) \in \Pi_k\right\}$$
4: $$N_{min_k} \leftarrow \min\left\{z : \text{for } \exists(\Psi_{max_k}, X_{max_k}, l, \pi_z^l),\ (\Psi_{max_k}, X_{max_k}, z, l, \pi_z^l) \in \Pi_k\right\}$$
5: $$D_{min_k} \leftarrow \min\left\{l : \text{for } \exists(\Psi_{max_k}, X_{max_k}, N_{min_k}, l, \pi_z^l),\ (\Psi_{max_k}, X_{max_k}, N_{min_k}, l, \pi_z^l) \in \Pi_k\right\}$$
6: $C_k \leftarrow \pi_z^l$ ▷ Where $\left(\Psi_{max_k}, X_{max_k}, N_{min_k}, D_{min_k}, \pi_z^l\right) \in \Pi_k$
7: **if** $|\Psi_j| > \Psi_{max_k} \vee (|\Psi_j| = \Psi_{max_k} \wedge X_j > X_{max_k}) \vee (|\Psi_j| = \Psi_{max_k} \wedge X_j = X_{max_k} \wedge N_{min_k} > j) \vee (N_{min_k} == j \wedge D_{min_k} > k')$ **then**
8: Send `CS` message, $\left(|\Psi_j|, X_j, j, k', \pi_j^{k'}\right)$, to one-hop neighbors on the subgraph of (i, k)
9: **for all** $(z, l) \in M_P[(i, k)]$ **do** ▷ Announce to delegated radios
10: Send `CS` message, $\left(|\Psi_j|, X_j, j, k', \pi_j^{k'}\right)$, to (z, l)
11: **end for**
12: **end if**
13: $\Pi_k \leftarrow \Pi_k \cup \left\{(|\Psi_j|, X_j, j, k', \pi_j^{k'})\right\}$
14: **end procedure**

our simulation results to assess the distributed scheme's performance. We use the same set of metrics (I_{ap}, I_{aph}, I_{awp} and R_{bc}) introduced in Section 6.3.1 to evaluate our scheme.

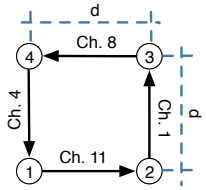
7.3.1 Validation Using Small Networks



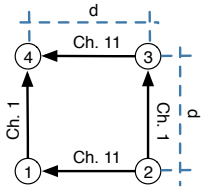
(a) 3 Nodes 2 Flows Linear



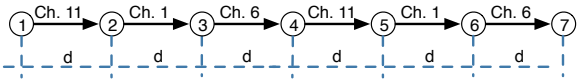
(b) 3 Nodes 3 Flows Circular



(c) 4 Nodes 4 Flows Circular



(d) 4 Nodes 4 Flows Multi-flow



(e) 7 Nodes 6 Flows Linear

Figure 7.4: Verification of the distributed scheme on small networks of two-radio nodes where $d_I = d_T$.

We ran simulations on five small networks (Figure 7.4) to validate the correctness of the proposed scheme. In Figure 7.4, circles represent two-radio nodes and arrows represent flows of equal magnitude between these nodes. The channels configured by the distributed scheme at the end of simulations are indicated atop the flow arrows. In each scenario, the interference range is equal to the transmission range. The simulation parameters used for these networks are given in Table 7.2. As evident from Figure 7.4, the proposed scheme is able to find optimal channel configurations for these networks.

7.3.2 Simulation Experiments

We run extensive simulation experiments to evaluate the flow-radio and channel assignment configurations that our DFRCA scheme produces. The configurations

Name	Value
d_I	1 d_T
d_D	1 d_T for 7.4(a)-7.4(d)
d_D	3 d_T for 7.4(e)
D	2
M	11
O_Δ	5

Table 7.2: Simulation parameters for Figure 7.4.

produced by DFRCA are compared against the *random* and *single-channel* configurations introduced in Section 6.3.2. Hence we compare three different types of configurations which are explained as follows:

1. *Single-channel configuration*: This is the same configuration as 6.3.2.1.
2. *Random configuration*: This is the same configuration as 6.3.2.2.
3. *DFRCA configuration*: This is a flow-radio coupling and channel configuration arrived at the end of the simulation process of our proposed DFRCA scheme.

In Figure 7.5, we observe the effects of the delegation range (d_D) on DFRCA's performance for a chain topology of 10 nodes. Delegation range (d_D) is a tunable parameter of DFRCA. When d_D is extended up to four or more times the transmission range (d_T), DFRCA yields an optimum solution. For smaller chains, DFRCA is able to find optimum solutions with smaller d_D. In backbone WMNs [1], where the traffic is routed towards a gateway node, a routing tree rooted at the gateway node is formed and such longer isolated chains are more common. However, if intra-mesh traffic does not concentrate on a special node as with backbone WMNs, a smaller d_D will suffice.

Figure 7.6 compares DFRCA against single-channel and random configuration schemes and shows how the metrics change as the network size increases when the number of available wireless channels (M) is 22. Relevant simulation parameters can be found in the second column of Table 7.3. For all four metrics, the single-channel configuration scheme has the worst performance and DFRCA

Name	Figure 7.6	Figures 7.7-7.10	Figures 7.11-7.14
N_d	2	2	2
d_I	1 d_T	1 d_T	1 d_T
d_D	1 d_T	1 d_T	1 d_T
D	2	2	2
M	22	11-55	22
O_Δ	5	5	1-9

Table 7.3: Simulation parameters for Figures 7.6-7.14.

has the best performance. For I_{ap}, DFRCA achieves up to 246% improvement with respect to the random configuration scheme and up to 298% improvement with respect to the single-channel configuration scheme, both for 16 nodes. For I_{awp}, the improvements are more pronounced: up to 819% with respect to the random configuration and more than 10 times with respect to the single-channel configuration, again both for 16 nodes. For the R_{bc} metric, DFRCA achieves up to 233% improvement for 16 nodes with respect to the random configuration and up to 867% improvement for 100 nodes with respect to the single-channel configuration. Figure 7.6 shows that, interestingly, the performance of the random configuration in terms of I_{aph} closely follows the performance of the single-channel configuration. The improvement achieved by DFRCA in terms of I_{aph} is 153% for 16 nodes and 145% for 100 nodes with respect to the single-channel configuration.

Figure 7.7 shows the averages of I_{ap} in relation to the increasing number of available wireless channels (M) over 50 topologies for node counts of $16, 36, 64$ and 100. The third column of Table 7.3 lists the relevant simulation parameters. The single-channel configuration scheme can make no use of the increasing number of wireless channels, whereas the random configuration scheme's performance increases as the number of available channels increases because it has more channels to select from. However, DFRCA can utilize an increasing number of wireless channels better than the random configuration even for large numbers of nodes and flows. It is important to note that the random configuration yields more or less the same performance as the single-channel configuration for 100 nodes in terms of I_{ap} when the number of available channels is 11 (as with IEEE 802.11 in the FCC domain).

Figure 7.8 reveals that the random configuration scheme performs worse than the single-channel configuration scheme in terms of I_{aph} when the number of available channels is 11. This result occurs because I_{aph} assumes full-duplex operation of the radios. The random configuration scheme can, to some extent, decouple flows better than the single-channel configuration scheme where all the radios in the network are on the same subgraph, however, the random configuration fails to operate those decoupled flows sufficiently spectrally away from each other for $M = 11, 22$. The single-channel configuration can yield less interference compared to the random configuration by coupling flows on the same radios. Our DFRCA, on the other hand, effectively decouples flows, operates them spectrally away from each other and can spatially reuse the channels, allowing improvements of at least 132% for $M = 11$ and at least 145% for $M = 55$ (both for 64 nodes) with respect to the random configuration. The improvements with DFRCA in terms of I_{aph} with respect to the single-channel configuration are at least 127% for $M = 11$ for 64 nodes and at least 153% for $M = 55$ for 100 nodes.

In Figure 7.9, we observe the effects of the increasing number of wireless channels on I_{awp}. The improvements gained with the distributed scheme are even more pronounced for the flow-magnitude weighted metric in all four cases because DFRCA is flow-aware.

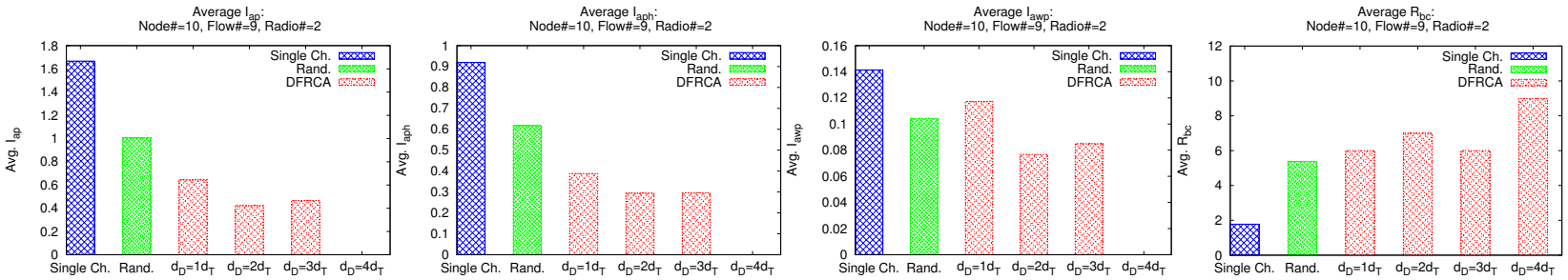
Figure 7.10 reveals that the proposed scheme can actually increase the average residual capacity in the network as the number of available channels increases. The random configuration can also increase the residual capacities, but in all four cases, DFRCA makes more intelligent use of the increase in the number of channels despite the fact that the number of available radios per node is kept constant. With 11 channels, there are at most three non-overlapping channels; with 22 channels there are five non-overlapping channels (channels $1, 6, 11, 16$ and 21) and with 33 channels there are seven non-overlapping channels. In all four cases, DFRCA can increase the performance for up to seven non-overlapping channels.

Next, we turn our attention to the relationships between the non-overlapping channel separation (O_Δ) and I_{ap}, I_{aph}, I_{awp} and R_{bc}. O_Δ is the minimum channel

separation needed to consider two wireless channels as non-overlapping (orthogonal). For IEEE 802.11b/g, when two channels are separated by at least five channels, they are considered to be non-overlapping [22], thus, channels 1, 6 and 11 of IEEE 802.11b/g are non-overlapping. In Figures 7.11, 7.12, 7.13 and 7.14, we observe the effects of increasing O_Δ on I_{ap}, I_{aph}, I_{awp} and R_{bc}, respectively, for a wireless technology that has 22 channels. When O_Δ is 1, all 22 channels are non-overlapping with respect to one another. When O_Δ is 9, there exist at most three non-overlapping channels amongst the 22 channels of the wireless technology: 1, 10 and 19. The simulation parameters used in these sets are given in the fourth column of Table 7.3.

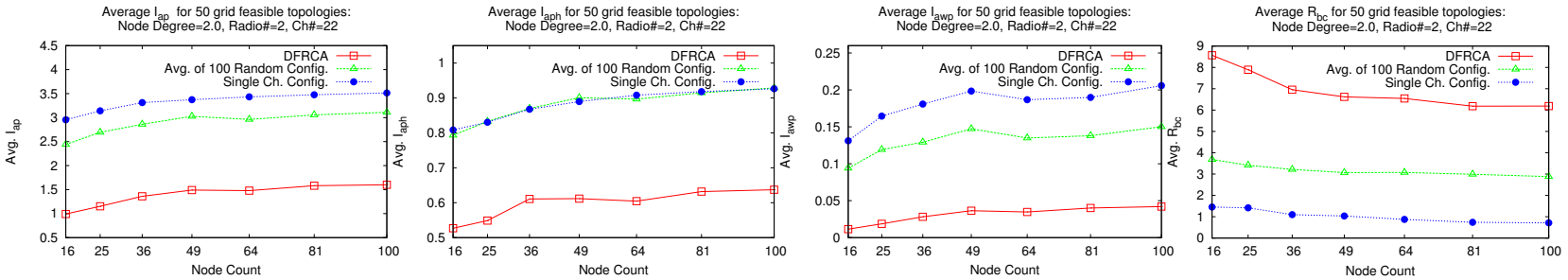
As Figure 7.11 reveals, I_{ap} increases for the random configuration scheme and DFRCA as O_Δ increases. Because the single-channel configuration uses only one channel, its performance is not affected by O_Δ. For 16 nodes and 16 flows in the network, the improvement gained by DFRCA with respect to the random configuration is 2.25 when $O_\Delta = 1$, and 2.09 when $O_\Delta = 9$. However, when there are 100 nodes and 100 flows in the network, the improvement gained by DFRCA over the random configuration scheme is 2.01 for $O_\Delta = 1$ and 1.6 for $O_\Delta = 9$. Hence, as the network grows in terms of node count and flow count, the number of available orthogonal channels becomes more important for DFRCA because it intelligently utilizes these orthogonal channels to reduce interference. This phenomenon can also be observed for I_{aph} and I_{awp} in Figures 7.12 and 7.13, respectively. I_{aph} increases faster for $O_\Delta > 5$ at $|N|= 64$ and $|N|= 100$ (see Figures 7.12(c) and 7.12(d), respectively). Similarly, I_{awp} increases faster for $O_\Delta > 5$ at $|N|= 64$ and $|N|= 100$ (see Figures 7.13(c) and 7.13(d), respectively) than at $|N|= 16$ or $|N|= 36$.

The observations made in Figures 7.11, 7.12 and 7.13 are verified in terms of the residual capacities in Figure 7.14. In all four cases, there is a linear decrease in R_{bc} as O_Δ increases for the random configuration scheme. However, R_{bc} decreases exponentially as O_Δ increases at $|N|= 36$, $|N|= 64$ and $|N|= 100$ with DFRCA (Figures 7.14(b), 7.14(c) and 7.14(d), respectively).



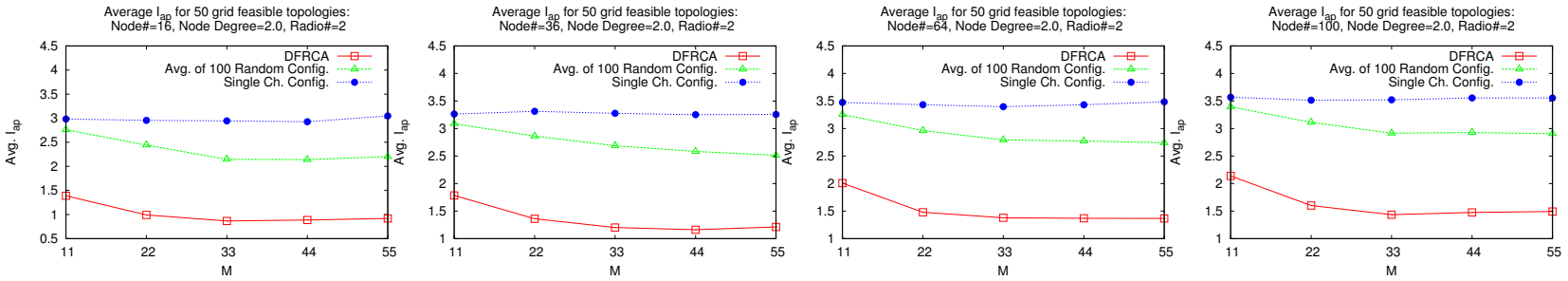
(a) I_{ap} (b) I_{aph} (c) I_{awp} (d) R_{bc}

Figure 7.5: Effects of the delegation range (d_D) on I_{ap}, I_{aph}, I_{awp}, and R_{bc} for a chain topology of 10 nodes.



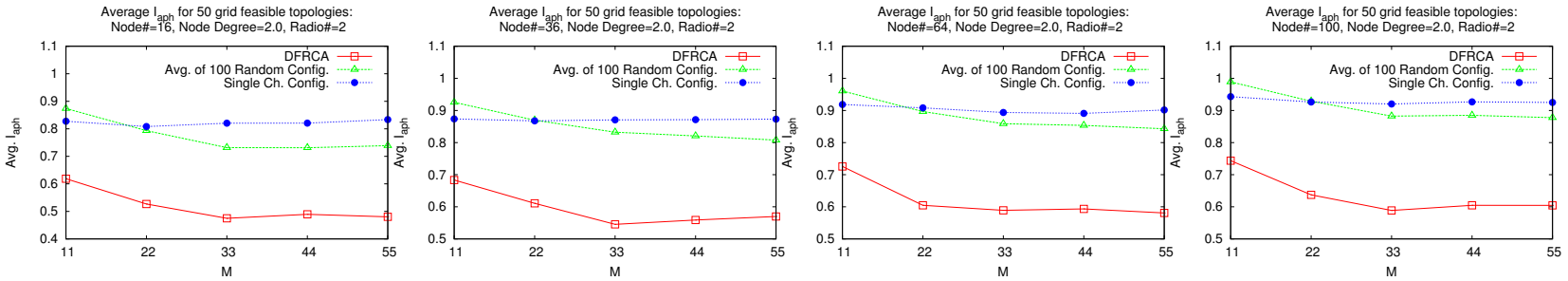
(a) I_{ap} (b) I_{aph} (c) I_{awp} (d) R_{bc}

Figure 7.6: Effects of the network size ($|N|$) on I_{ap}, I_{aph}, I_{awp}, and R_{bc}.



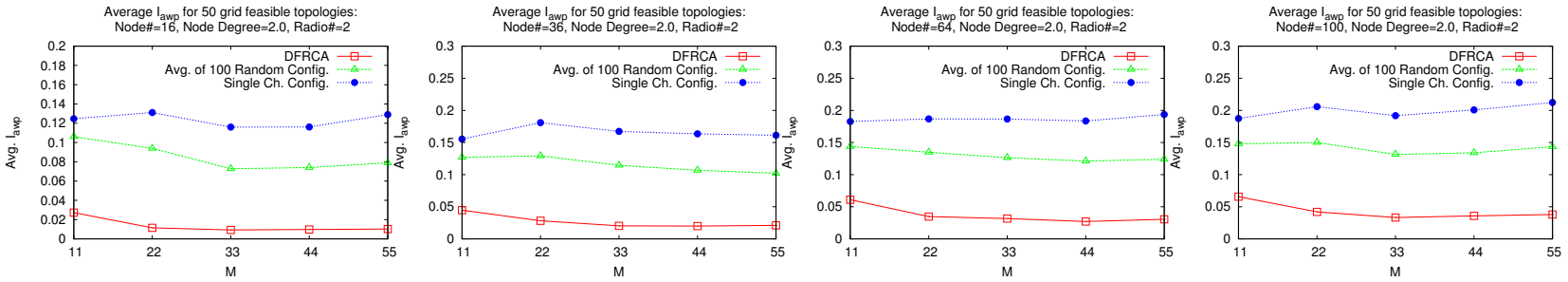
(a) $|N|= 16$ (b) $|N|= 36$ (c) $|N|= 64$ (d) $|N|= 100$

Figure 7.7: Effects of the number of available wireless channels (M) on I_{ap} for different network sizes.



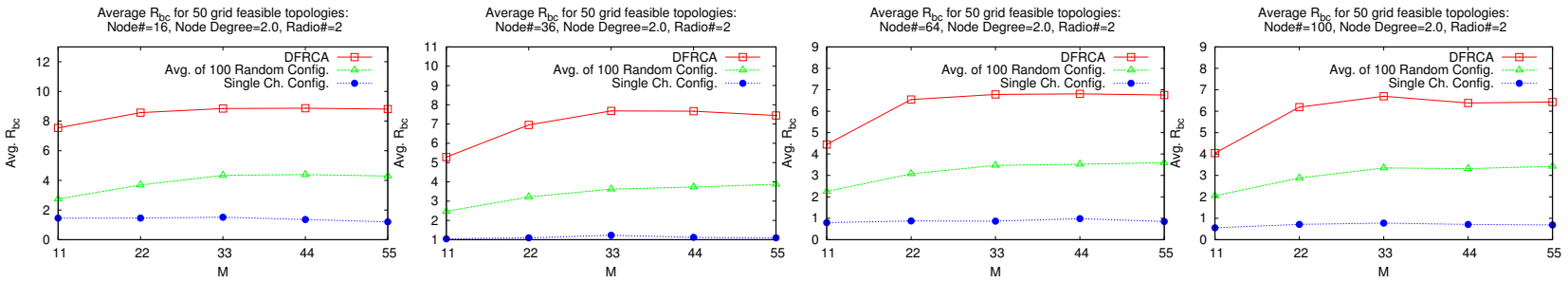
(a) $|N|= 16$ (b) $|N|= 36$ (c) $|N|= 64$ (d) $|N|= 100$

Figure 7.8: Effects of the number of available wireless channels (M) on I_{aph} for different network sizes.



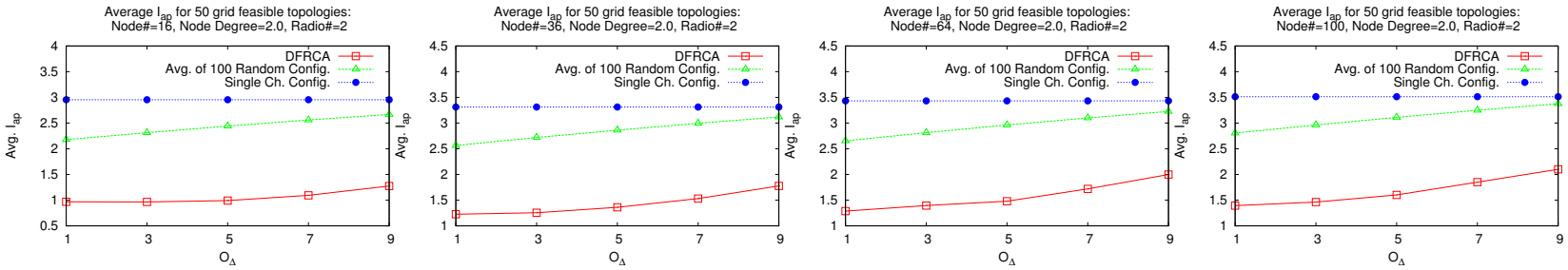
(a) $|N|= 16$ (b) $|N|= 36$ (c) $|N|= 64$ (d) $|N|= 100$

Figure 7.9: Effects of the number of available wireless channels (M) on I_{awp} for different network sizes.



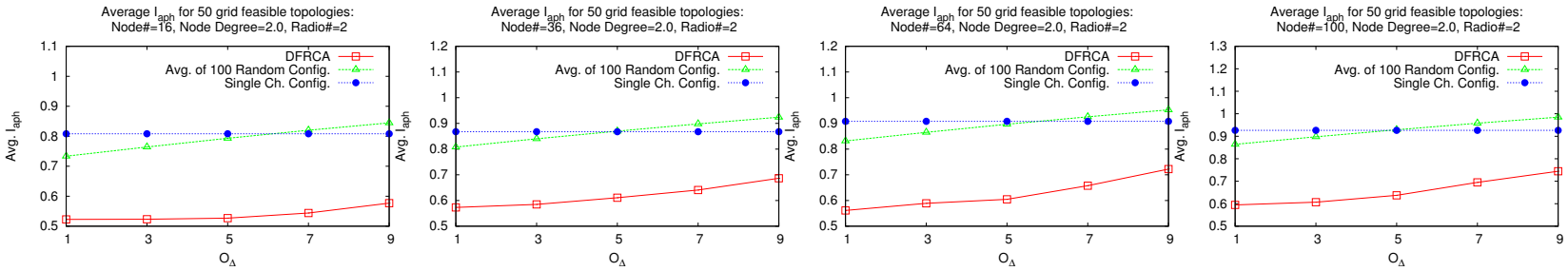
(a) $|N|= 16$ (b) $|N|= 36$ (c) $|N|= 64$ (d) $|N|= 100$

Figure 7.10: Effects of the number of available wireless channels (M) on R_{bc} for different network sizes.



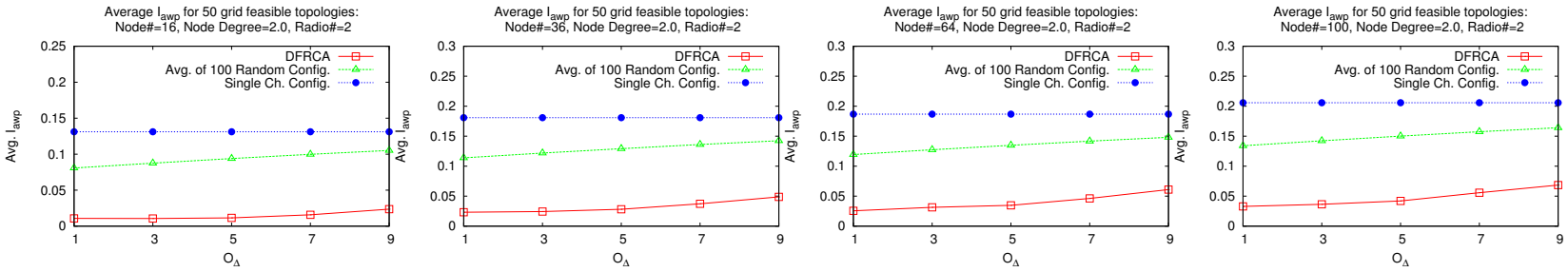
(a) $|N|= 16$ (b) $|N|= 36$ (c) $|N|= 64$ (d) $|N|= 100$

Figure 7.11: Effects of the non-overlapping channel separation (O_Δ) on I_{ap} for different network sizes for $M = 22$.



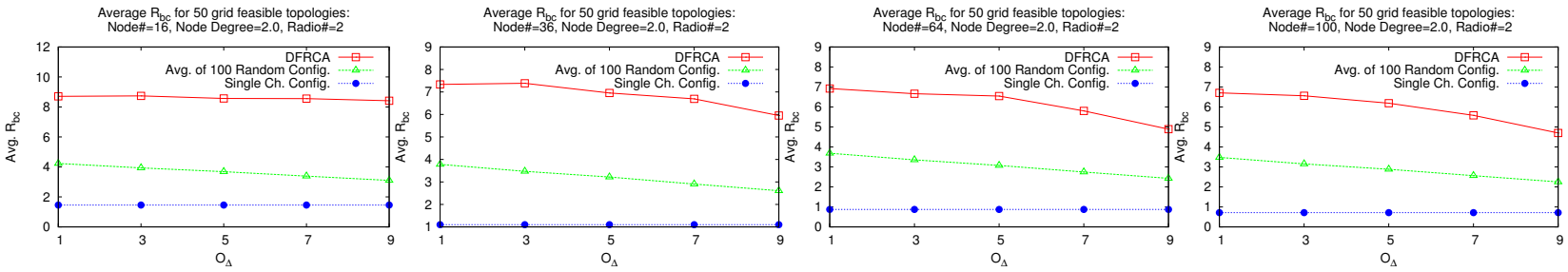
(a) $|N|= 16$ (b) $|N|= 36$ (c) $|N|= 64$ (d) $|N|= 100$

Figure 7.12: Effects of the non-overlapping channel separation (O_Δ) on I_{aph} for different network sizes when $M = 22$.



(a) $|N|= 16$ (b) $|N|= 36$ (c) $|N|= 64$ (d) $|N|= 100$

Figure 7.13: Effects of the non-overlapping channel separation (O_Δ) on I_{awp} for different network sizes when $M = 22$.



(a) $|N|= 16$ (b) $|N|= 36$ (c) $|N|= 64$ (d) $|N|= 100$

Figure 7.14: Effects of the non-overlapping channel separation (O_Δ) on R_{bc} for different network sizes when $M = 22$.

7.4 Summary

Flow-radio coupling in multi-radio WMNs has a prominent impact on channel assignment because of the physical constraints of the radios. Jointly addressing the flow-radio assignment and channel assignment problems therefore has the potential to increase WMN capacity by mitigating inter-flow and multi-hop intra-flow interference.

The DFRCA protocol we propose effectively addresses these two problems in a joint manner. As the simulation results show, our DFRCA increases the residual capacities of the receivers and mitigates interference significantly in the contexts of half-duplex as well as full-duplex radio technologies. We evaluate DFRCA performance using different radio and interference models and with solid metrics assessing various aspects of a WMN. Our DFRCA achieves up to eight times improvement in terms of the average traffic-weighted protocol interference with respect to the random configuration scheme and up to 10 times improvement with respect to the single-channel configuration scheme. When the average residual capacities of the receivers are considered, our DFRCA achieves over twofold improvement with respect to the random configuration and over eightfold improvement with respect to the single-channel configuration.

The proposed DFRCA can significantly enhance the utilization of the radio resources, such as the available spectrum and radios. Using our novel concept of disjoint subgraphs of radios, the DFRCA effectively decouples flows and operates them as spectrally far as possible from each other. This DFRCA also spatially reuses channels by grouping *non-interfering subgraphs* in colour classes and assigning channels to these colour classes.

Chapter 8

Conclusions and Future Work

Wireless mesh networking has attracted the attention of the research community with its promises and the broad range of problems it poses. Multi-radio multi-channel WMNs promise to increase the available capacity, however, they pose new and elaborate research problems.

This thesis is a modest attempt to address some of the critical issues in the context of the multi-radio multi-channel WMNs. Although the channel assignment problem has been extensively studied by the research community, the equally important flow-radio assignment problem has been overlooked. Also, majority of the literature on channel assignment considers only non-overlapping channels, partly because of the fact that the interaction between overlapping channels and the effects of this interaction on the various layers of the network have not been fully understood. This thesis aims to contribute to the general understanding and awareness of these complex interactions and affects by proposing novel theoretical and practical tools.

To investigate how the consecutive hops of a multi-hop flow interact with each other when they are on the same channel and when they are on different (overlapping or orthogonal) channels, we have established an indoor multi-hop multi-channel WMN testbed. We have designed a novel, cost-effective multi-radio

node architecture, which addresses some key issues previously raised by the research community. Using these multi-radio relay nodes, we have drawn important conclusions on the achievable multi-hop TCP and UDP goodput and also on other network layer metrics. Our findings after these experiments remarkably show that CSMA/CA misbehaves in certain situations where overlapping channels are used in consecutive hops, seriously degrading the network performance.

To address the channel assignment problem using overlapping channels, we needed a quantitative model of the interference between the channels of a wireless communication technology. To develop such a model, we have made measurements of the power spectral densities of the IEEE 802.11b signals using the multi-radio nodes of the testbed. Based on these measurements, we have proposed two measurement-based methods for the calculation of the interference factors (I-factors). Due to the technology-independence of these methods, they can also be used to calculate the interference factors between the channels of two different communication standards. We have reported our I-factor calculations for the interference between the IEEE 802.11b DSSS channels and between the IEEE 802.15.4 and 802.11b channels.

To further analyze the effects of flow-radio assignment (how flows are assigned to radios of a multi-radio node) on the total interference and the effects of adjacent channel interference on link capacities, we have developed two optimization models. These models also incorporate the effects of an idealized MAC protocol on achievable capacities. These models allow us to gain further insight into the joint flow-radio and channel assignment problem.

Finally, we have addressed the joint flow-radio and channel assignment problems by a set of centralized and distributed algorithms. To evaluate the performance of these algorithms, we have also proposed a set of solid metrics assessing the average protocol interference, average physical interference, and the average residual capacities of the receivers. We have been able to achieve substantial improvements over random and single channel configurations on random topologies in terms of these metrics.

There are various dimensions towards which the research presented in this

thesis can be extended. As indicated in Chapter 5, the optimization models presented there can be converted to (mixed) integer linear programming models (with the necessary simplifications or assumptions). Another dimension is to extend the work presented in Chapters 5-7 to incorporate adaptive modulation and coding in their formulations.

Concerning the centralized and distributed schemes proposed in Chapters 6 and 7, a valuable effort would be to implement them on BilMesh and to fine-tune their parameters for real-world situations. Concerning the DFRCA scheme (of Chapter 7), the distributed protocol definition can be extended to include the execution schedules and the necessary protocol signaling to trigger those executions.

Bibliography

[1] I. F. Akyildiz, X. Wang, and W. Wang, "Wireless Mesh Networks: A Survey," *Computer Networks*, vol. 47, pp. 445–487, Mar. 2005.

[2] T. S. Rappaport, *Wireless Communications Principle and Practice*. Prentice Hall Ptr, Second ed., 2002.

[3] "The Wi-Fi Alliance." Available from: <http://www.wi-fi.org> Last Access: Oct. 2011.

[4] "The WiMAX Forum." Available from: <http://www.wimaxforum.org> Last Access: Oct. 2011.

[5] "ZigBee Specification," *ZigBee Document Ver. 053474r06*, 2008.

[6] J. So and N. Vaidya, "Multi-Channel MAC for Ad Hoc Networks: Handling Multi-Channel Hidden Terminals Using A Single Transceiver," *Proceedings of the* 5^{th} *ACM International Symposium on Mobile Ad hoc Networking and Computing (MobiHoc'04)*, pp. 222–233, 2004.

[7] P. Bahl, R. Chandra, and J. Dunagan, "SSCH: Slotted Seeded Channel Hopping for Capacity Improvement in IEEE 802.11 Ad-Hoc Wireless Networks," *Proceedings of the* 10^{th} *Annual International Conference on Mobile Computing and Networking (MobiCom'04)*, 2004.

[8] A. Adya, P. Bahl, J. Padhye, A. Wolman, and L. Zhou, "A Multi-Radio Unification Protocol for IEEE 802.11 Wireless Networks," *Proceedings of the* 1^{st} *International Conference on Broadband Networks (BroadNets'04)*, 2004.

[9] J. Bicket, D. Aguayo, S. Biswas, and R. Morris, "Architecture and Evaluation of an Unplanned 802.11b Mesh Network," *Proceedings of the* 11^{th} *Annual International Conference on Mobile Computing and Networking (MobiCom'05)*, pp. 31–42, 2005.

[10] J. C. Bicket, "Bit-rate Selection in Wireless Networks," *Wireless Networks*, pp. 1–50, 2005.

[11] "Microsoft Mesh Connectivity Layer (MCL) Software." Available from: <`http://research.microsoft.com/mesh`> Last Access: Oct. 2011.

[12] R. Draves, J. Padhye, and B. Zill, "Routing in Multi-radio, Multi-hop Wireless Mesh Networks," *Proceedings of the* 10^{th} *Annual International Conference on Mobile Computing and Networking (MobiCom'04)*, vol. 04, pp. 114–128, 2004.

[13] Y. Amir, C. Danilov, M. Hilsdale, R. Mus loiu Elefteri, and N. Rivera, "Fast Handoff for Seamless Wireless Mesh Networks," in *Proceedings of the* 4^{th} *International Conference on Mobile Systems, Applications and Services (MobiSys'06)*, pp. 83–95, ACM, 2006.

[14] "The Spines Overlay Network." Available from: <`http://www.spines.org`> Last Access: Oct. 2011.

[15] Y. Amir and C. Danilov, "Reliable Communication in Overlay Networks," *Proceedings of the IEEE International Conference on Dependable Systems and Networks (DSN'03)*, pp. 511–520, June 2003.

[16] Y. Amir, C. Danilov, R. Musuualoiu-Elefteri, and N. Rivera, "The SMesh Wireless Mesh Network," *Science And Technology*, vol. 28, no. 3, pp. 1–4, 2010.

[17] J. Robinson, K. Papagiannaki, C. Diot, X. Guo, and L. Krishnamurthy, "Experimenting with a Multi-Radio Mesh Networking Testbed," in *The* 1^{st} *International Workshop on Wireless Network Measurements (WiNMee'05)*, pp. 1–6, 2005.

[18] C. Zhang, K. Kowalik, and M. Davis, "An Experimental Study of the Impact of Using Multi-Radio in WLAN Mesh Networks," in *Proceedings of the* 5^{th} *International Conference on Wireless Communications, Networking and Mobile Computing (WiCom'09)*, pp. 1–4, IEEE, 2009.

[19] U. Madhow, *Fundamentals of Digital Communication*. Cambridge University Press, First ed., 2008.

[20] M. Gast, *802.11 Wireless Networks: The Definitive Guide*. O'Reilly Media, Inc., Second ed., 2005.

[21] A. Goldsmith, *Wireless Communications*. Cambridge university press, First ed., 2005.

[22] "IEEE Std 802.11™-2007, IEEE Standard for Information Technology–Telecommunications and Information Exchange Between Systems–LANs and MANs–Specific Requirements–Part 11: WLAN MAC and PHY Specifications," June 2007.

[23] A. Mishra, E. Rozner, S. Banerjee, and W. Arbaugh, "Using Partially Overlapped Channels in Wireless Meshes," *Wimesh, Santa Clara*, vol. 26, 2005.

[24] A. Mishra, E. Rozner, S. Banerjee, and W. Arbaugh, "Exploiting Partially Overlapping Channels in Wireless Networks: Turning a Peril into an Advantage," in *Proceedings of the* 5^{th} *ACM SIGCOMM Conference on Internet Measurement*, pp. 311–316, USENIX Association, 2005.

[25] A. Mishra, V. Shrivastava, S. Banerjee, and W. Arbaugh, "Partially Overlapped Channels Not Considered Harmful," *ACM SIGMETRICS Performance Evaluation Review*, vol. 34, pp. 63–74, June 2006.

[26] E. G. Villegas, E. L. Aguilera, R. Vidal, and J. Paradells, "Effect of Adjacent-Channel Interference in IEEE 802.11 WLANs," in 2^{nd} *International Conference on Cognitive Radio Oriented Wireless Networks and Communications (CrownCom'07)*, 2007.

[27] H. Zimmermann, "Open Systems Interconnection," *IEEE Transactions on Communications*, vol. 28, no. 4, pp. 425–432, 1980.

[28] Z. Feng and Y. Yang, "How Much Improvement Can We Get From Partially Overlapped Channels?," *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC'08)*, pp. 2957–2962, Mar. 2008.

[29] Z. Feng and Y. Yang, "Characterizing the Impact of Partially Overlapped Channels on the Performance of Wireless Networks," *IEEE Global Telecommunications Conference (GLOBECOM'08)*, pp. 1–6, 2008.

[30] G. Zhou, J. A. Stankovic, and S. H. Son, "Crowded Spectrum in Wireless Sensor Networks," *IEEE EmNets*, 2006.

[31] P. Fuxjäger, D. Valerio, and F. Ricciato, "The Myth of Non-Overlapping Channels: Interference Measurements in IEEE 802.11," *Fourth Annual Conference on Wireless on Demand Network Systems and Services*, pp. 1–8, Jan. 2007.

[32] Y. Yao, A. U. H. Sheikh, and S. Cheng, "Near/far Effects on Packet Radio Networks with Direct-Sequence Spread-Spectrum Signaling," in *IEEE Pacific Rim Conference on Communications, Computers and Signal Processing*, 1989.

[33] M. Petrova, L. Wu, P. Mähönen, and J. Riihijärvi, "Interference Measurements on Performance Degradation Between Colocated IEEE 802.11g/n and IEEE 802.15.4 Networks," *Proceedings of the* 6^{th} *International Conference on Networking (ICN'07)*, 2007.

[34] J. Li, C. Blake, D. S. De Couto, H. I. Lee, and R. Morris, "Capacity of Ad Hoc Wireless Networks," in *Proceedings of the* 7^{th} *Annual International Conference on Mobile Computing and Networking (MobiCom'01)*, pp. 61–69, ACM, 2001.

[35] J. Jun and M. Sichitiu, "The Nominal Capacity of Wireless Mesh Networks," *IEEE Wireless Communications*, vol. 10, no. 5, pp. 8–14, 2003.

[36] P.-J. Wan, X. Xu, Z. Wang, S. Tang, and Z. Wan, "Stability Analyses of Longest-Queue-First Link Scheduling in MC-MR Wireless Networks," in *Proceedings of the* 13^{th} *ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc'12)*, pp. 45–54, ACM, 2012.

[37] A. Raniwala, K. Gopalan, and T. Chiueh, "Centralized Channel Assignment and Routing Algorithms for Multi-Channel Wireless Mesh Networks," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 8, no. 2, pp. 50–65, 2004.

[38] M. Alicherry, R. Bhatia, and L. E. Li, "Joint Channel Assignment and Routing for Throughput Optimization in Multi-radio Wireless Mesh Networks," in *Proceedings of the* 11^{th} *Annual International Conference on Mobile Computing and Networking (MobiCom'05)*, pp. 58–72, ACM, 2005.

[39] A. Rad and V. Wong, "Joint Optimal Channel Assignment and Congestion Control for Multi-channel Wireless Mesh Networks," in *IEEE International Conference on Communications (ICC'06)*, vol. 5, pp. 1984–1989, 2006.

[40] A. Raniwala and T. Chiueh, "Architecture and Algorithms for an IEEE 802.11-Based Multi-Channel Wireless Mesh Network," in 24^{th} *IEEE International Conference on Computer Communications (INFOCOM'05)*, vol. 3, pp. 2223–2234, 2005.

[41] V. Bukkapatanam, A. Franklin, and C. Murthy, "Using Partially Overlapped Channels for End-to-End Flow Allocation and Channel Assignment in Wireless Mesh Networks," in *IEEE International Conference on Communications (ICC'09)*, pp. 1–6, 2009.

[42] A. H. Rad and V. W. Wong, "Partially Overlapped Channel Assignment for Multi-Channel Wireless Mesh Networks," in *IEEE International Conference on Communications (ICC'07)*, pp. 3770–3775, 2007.

[43] K. N. Ramachandran, E. M. Belding, K. C. Almeroth, and M. M. Buddhikot, "Interference-Aware Channel Assignment in Multi-Radio Wireless Mesh Networks," in 25^{th} *IEEE International Conference on Computer Communications (INFOCOM'06)*, vol. 6, pp. 1–12, 2006.

[44] H. Skalli, S. Ghosh, S. K. Das, L. Lenzini, and M. Conti, "Channel Assignment Strategies for Multiradio Wireless Mesh Networks: Issues and Solutions," *IEEE Communications Magazine*, vol. 45, no. 11, pp. 86–95, 2007.

[45] D. Wang, P. Lv, Y. Chen, and M. Xu, "POCAM: Partially Overlapped Channel Assignment in Multi-Radio Multi-Channel Wireless Mesh Network," in 11^{th} *International Symposium on Communications and Information Technologies (ISCIT'11)*, pp. 188–193, 2011.

[46] M. Hoque, X. Hong, and F. Afroz, "Multiple Radio Channel Assignment Utilizing Partially Overlapped Channels," in *IEEE Global Telecommunications Conference (GLOBECOM'09)*, pp. 1–7, 2009.

[47] A. Subramanian, H. Gupta, S. Das, and J. Cao, "Minimum Interference Channel Assignment in Multiradio Wireless Mesh Networks," *IEEE Transactions on Mobile Computing*, vol. 7, no. 12, pp. 1459–1473, 2008.

[48] H. Wu, F. Yang, K. Tan, J. Chen, Q. Zhang, and Z. Zhang, "Distributed Channel Assignment and Routing in Multiradio Multichannel Multihop Wireless Networks," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 11, pp. 1972–1983, 2006.

[49] X. Lin and S. Rasool, "A Distributed Joint Channel-Assignment, Scheduling and Routing Algorithm for Multi-Channel Ad Hoc Wireless Networks," in 26^{th} *IEEE International Conference on Computer Communications (INFOCOM'07)*, pp. 1118–1126, IEEE, 2007.

[50] A. Dhananjay, H. Zhang, J. Li, and L. Subramanian, "Practical, Distributed Channel Assignment and Routing in Dual-radio Mesh Networks," *SIGCOMM Computer Communication Review*, vol. 39, pp. 99–110, Aug. 2009.

[51] M. Shin, S. Lee, and Y.-a. Kim, "Distributed Channel Assignment for Multiradio Wireless Networks," in *IEEE International Conference on Mobile Adhoc and Sensor Systems (MASS'06)*, pp. 417–426, IEEE, 2006.

[52] A. Naveed, S. Kanhere, and S. Jha, "Topology Control and Channel Assignment in Multi-Radio Multi-Channel Wireless Mesh Networks," in *IEEE International Conference on Mobile Adhoc and Sensor Systems (MASS'07)*, pp. 1–9, 2007.

[53] B.-J. Ko, V. Misra, J. Padhye, and D. Rubenstein, "Distributed Channel Assignment in Multi-Radio 802.11 Mesh Networks," in *IEEE Wireless Communications and Networking Conference (WCNC'07)*, pp. 3978–3983, 2007.

[54] "RFC3626: Optimized Link State Routing Protocol (OLSR)," 2003.

[55] "olsrd Project Home Page." Available from: <http://www.olsr.org/> Last Access: Oct. 2011.

[56] "OpenWRT, Linux for Embedded Devices." Available from: <http://openwrt.org> Last Access: Oct. 2011.

[57] "Apache Geronimo Project Home Page." Available from: <http://geronimo.apache.org/> Last Access: Oct. 2011.

[58] "Linksys WAP54G Product Support Page." Available from: <http://homesupport.cisco.com/en-eu/support/routers/WAP54G> Last Access: Oct. 2011.

[59] "Broadcom BCM4318E Product Home Page." Available from: <http://www.broadcom.com/products/Wireless-LAN/Wi-Fi-Phone-Solutions/BCM4318E> Last Access: Oct. 2011.

[60] "Linksys WRT54GL Product Support Page." Available from: <http://homesupport.cisco.com/en-eu/support/routers/WRT54GL> Last Access: Oct. 2011.

[61] "Broadcom BCM5352EL Product Home Page." Available from: <http://www.broadcom.com/products/Wireless-LAN/802.11-Wireless-LAN-Solutions/BCM5352EL> Last Access: Oct. 2011.

[62] "RFC792: Internet Control Message Protocol (ICMP)," pp. 11–12, 1981.

[63] "Iperf: The TCP/UDP Bandwidth Measurement Tool." Last Access: Dec. 2012.

[64] "RFC3550: RTP: A Transport Protocol for Real-Time Applications," pp. 39–40, 2003.

[65] J. Nachtigall, A. Zubow, and J.-P. Redlich, "The Impact of Adjacent Channel Interference in Multi-Radio Systems Using IEEE 802.11," in *International Wireless Communications and Mobile Computing Conference (IWCMC'08)*, pp. 874–881, 2008.

[66] P. Duarte, Z. Fadlullah, K. Hashimoto, and N. Kato, "Partially Overlapped Channel Assignment on Wireless Mesh Network Backbone," in *IEEE Global Telecommunications Conference (GLOBECOM'10)*, pp. 1–5, 2010.

[67] S.-H. Lim, C. Kim, Y.-B. Ko, and N. Vaidya, "Efficient Multicasting for Multi-channel Multi-interface Wireless Mesh Networks," in *IEEE Military Communications Conference (MILCOM'09)*, pp. 1–7, 2009.

[68] A. R. Ulucinar, I. Korpeoglu, and E. Karasan, "A Novel Measurement-based Approach for Modeling and Computation of Interference Factors for Wireless Channels," *EURASIP Journal on Wireless Communications and Networking*, 2013.

[69] "Yellowjacket-Tablet Wi-Fi Analyzer." Available from: <`http://www.bvsystems.com/Products/WLAN/YJ-TABLET/yj-tablet.htm`> Last Access: Jun. 2012.

[70] "Maxim MAX2820, MAX2820A, MAX2821, MAX2821A 2.4GHz 802.11b Zero-IF Transceivers Data Sheet." Available from: <`http://datasheets.maxim-ic.com/en/ds/MAX2820-MAX2821A.pdf`> Last Access: Jul. 2012.

[71] "Memsic Wireless Modules Product Page." Available from: <`http://www.memsic.com/products/wireless-sensor-networks/wireless-modules.html`> Last Access: Jun. 2012.

[72] "IEEE Std 802.15.4™-2006, IEEE Standard for Information Technology–Telecommunications and information exchange between systems– Local and metropolitan area networks–Specific requirements–Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs)," September 2006.

[73] P. M. Shankar, *Introduction to Wireless Systems*. John Wiley and Sons, First ed., 2002.

[74] "ILOG CPLEX 11.0 User's Manual," 2007.

[75] K. C.-J. Lin, S. Gollakota, and D. Katabi, "Random Access Heterogeneous MIMO Networks," in *Proceedings of the ACM SIGCOMM 2011 Conference*, pp. 146–157, ACM, 2011.

[76] R. Diestel, *Graph Theory*. Graduate Texts In Mathematics, Springer Verlag, Second ed., 2000.

[77] "CSIM for Java." Available from: <`http://www.mesquite.com/csimforjava`> Last Access: Mar. 2013.