Distributed Channel Assignment for Interference-Aware Wireless Mesh Networks

Dissertation

zur Erlangung des akademischen Grades Dr. rer. nat.

im Fach Informatik

eingereicht an der

Mathematisch-Naturwissenschaftlichen Fakultät II
der Humboldt-Universität zu Berlin

von

Dipl.-Inform. Felix Shzu-Juraschek

Präsident der Humboldt-Universität zu Berlin

Prof. Dr. Jan-Hendrik Olbertz

Dekan der Mathematisch-Naturwissenschaftlichen Fakultät II

Prof. Dr. Elmar Kulke

Gutachter

1. Prof. Dr. rer. nat. Mesut Güneş
2. Prof. Dr. rer. nat. Jens-Peter Redlich
3. Dr. Ahmet Sekiorciouglu

Tag der Verteidigung: 9. Mai 2014

Abstract

The rapid success of wireless radio communication has shaped several aspects of our social and work lives. This success is still far from reaching its full potential: with the advent of the Internet of Things, future trends indicate an exponential increase of wireless devices for machine-to-machine communication. Due to the broadcast nature of the shared medium, wireless transmissions are potentially received by all network stations in the communication range of the sender. With an unsynchronized medium access, multiple transmissions may be active at the same time and thus interfere with each other. In consequence, multiple transmissions may collide at the receiver side and can not be properly decoded [1].

For this reason, protocols have been developed on the MAC layer to synchronize the medium access and thus reduce interference effects [2]. One of these approaches in wireless mesh networks (WMNs) is channel assignment. The idea of channel assignment is to minimize the network-wide interference by utilizing non-overlapping channels for otherwise interfering wireless transmissions. This is feasible, since wireless mesh routers are usually equipped with multiple radios and commonly used wireless network technologies, such as IEEE 802.11, provide multiple non-overlapping channels [3]. Since IEEE 802.11 operates in the unlicensed frequency spectrum, the dense distribution of private and commercial network deployments of WLANs in urban areas poses a new challenge. Co-located networks compete for the wireless medium, thus decreasing the achievable network performance in terms of throughput and latency. Therefore, an important issue for efficient channel assignment is to also address *external interference*. This task is not trivial, since the external networks and devices are not under the control of the network operator and their radio activity is therefore hard to capture.

The contributions of this dissertation comprise the design, implementation, and validation of models and algorithms to enable wireless multi-hop networks to become interference-aware. This includes a measurement-based interference model that determines efficiently and accurately the interference relationships among network nodes in large-scale network deployments. A distributed channel assignment algorithm has been developed that considers external sources of interference. The overall solution has been experimentally validated in a large-scale wireless multi-hop multi-radio testbed and has significantly increased the network performance with regard to the network capacity.

Zusammenfassung

Der große Erfolg drahtloser Kommunikationstechnologien hat unsere Kommunikationsmöglichkeiten im Arbeitsleben und im sozialen Umfeld revolutioniert. Diese Revolution setzt sich weiter fort – Mit der Realisierung des Internet der Dinge steigt die Verbreitung von untereinander drahtlos kommunizierenden Geräten rasant an. Die Besonderheit der drahtlosen Kommunikation gegenüber den drahtgebundenen Netzwerken liegt im drahtlosen Übertragungsmedium. Aufgrund der Broadcast-Eigenschaft des Übertragungsmediums werden Nachrichten potentiell von allen Netzwerkstationen empfangen, welche sich in der Übertragungsreichweite des Senders aufhalten. Als Konsequenz können bei einem unsynchronisierten Medienzugriff mehrere Nachrichten beim Empfänger kollidieren und nicht korrekt empfangen werden. Dieses Phänomen wird auch als Interferenz bezeichnet [1].

Um solche Interferenzen zu vermeiden, wurden spezielle Protokolle für den Medienzugriff in drahtlosen Netzen entwickelt [2]. Ein solcher Ansatz für drahtlose Maschennetze ist die verteilte Kanalzuweisung. Bei der verteilten Kanalzuweisung werden sich nichtüberlappende Kanäle im verfügbaren Frequenzspektrum für Übertragungen verwendet, die auf dem gleichen Kanal Interferenzen erzeugen würden. Dieser Ansatz ist möglich, da die verwendeten Funktechnologien, wie zum Beispiel IEEE 802.11 (WLAN), mehrere nichtüberlappende Kanäle bereitstellen [3]. Aufgrund der großen Verbreitung von IEEE 802.11, ist eine hohe Dichte von privaten wie kommerziellen Netzen im urbanen Raum die Norm. Diese räumlich überlappenden Netze konkurrieren um den Medienzugriff und die dadurch entstehende Interferenz kann die Netzleistung mindern. Daher ist es für die Leistung von Kanalzuweisungsalgorithmen von großer Bedeutung, die Aktivität der externen Netze mit einzubeziehen.

Die Leistung der vorgelegten Arbeit umfasst das Design, die Implementierung und Validierung von Modellen und Algorithmen zur Reduzierung von Interferenzen in drahtlosen Maschennetzen. Die Arbeit beinhaltet die Entwicklung eines Messungs-basierten Interferenzmodells, mit dem Interferenzabhängigkeiten der Maschenrouter untereinander effizient bestimmt werden können. Weiterhin wurde ein Algorithmus für die verteilte Kanalzuweisung entwickelt, der die Aktivität von externen Netzen berücksichtigt. Die Gesamtlösung wurde in einem großen drahtlosen Maschennetz implementiert und experimentell validiert. Die Ergebnisse haben gezeigt, dass die Gesamtlösung die Netzleistung in Bezug auf die Netzkapazität signifikant steigert.

I dedicate this dissertation to my wife Bettina, my parents Brunhilde and Klaus, and my grandmother Margarete.

I am incredibly grateful for Bettina's unconditional love and support. My parents deserve my sincerest gratefulness for their kind support and encouragement during my studies. Finally, I want to express my gratitude to my grandmother Margarete for her love and patience in my earliest years.

This dissertation would not have been possible without the support of so many people in many different ways. I owe my deepest appreciation to Prof. Dr. rer. nat. Mesut Güneş who accompanied me throughout the work of the dissertation and for sharing his great expertise of scientific research. Furthermore, I want to thank him for his constant support and the valuable feedback.

I am also deeply grateful to Prof. Dr. rer. nat. Jens-Peter Redlich for supervising my dissertation as part of the METRIK research training group. My deepest gratitude also goes to Dr. Ahmet Sekercioglu for supervising my dissertation and for his kind hospitality on my exciting and unforgettable research stay in his group at the Monash University in Melbourne. I would like to thank Prof. Dr. sc. Joachim Fischer for letting me join the METRIK research training group. The valuable feedback I got on the regular evaluation workshops have accelerated the completion of this dissertation and his generous efforts allowed me to disseminate my research results on several conferences.

I would also like to thank my colleagues at Freie Universität Berlin and Humboldt Universität Berlin for the feedback and encouragement from countless conversations. I especially thank my office mate Bastian Blywis for the fun 4 years. Bastian's expertise and even more his euphoric ways of sharing his knowledge were an inspiration. I am also grateful that I was a able to participate in the initial deployment of the DES-Testbed. The passionate team effort created one of the largest experimental facilities which was very exciting to have at our hands. I would like to thank Oliver Hahm for almost 10 years of mutual studies, work, and friendship. And especially for our daily bike rides through Berlin with contemplating discussions of our research and life.

I want to thank all the students of the DES working group, which I had the honor to supervise. Their efforts and even more the discussions of their results were very valuable for the direction of the work in this dissertation. Most notably, I want to thank Matthias Philipp and Simon Seif for their support in the channel assignment experiments.

I also want to thank my co-founders of Spreebytes, namely Heiko Will, Eike Send, Stephan Adler, and Kaspar Schleiser. The mutual passion for exciting IT projects served as a great inspiration to me and I am grateful for the growing friendships.

Finally, I thank the DAAD for granting me a scholarship for my guest researcher stay at the Monash University in Melbourne, Australia.

Berlin, November 2013
Felix Shzu-Juraschek

Contents

List of Figures

List of Tables

CHAPTER 1

Introduction

The rapid success of wireless radio communication has shaped several aspects of our social and work lives. In developed countries cell and smart phones are ubiquitous, allowing people to communicate with each other at almost any time from anywhere [4]. Laptops and tablet computers for work or private use are nowadays equipped with multiple wireless technologies such as WiFi [3] and Bluetooth [5]. The rise of wireless technologies leads to an increase of wireless traffic at enormous rates - the global mobile data traffic grew about 70% in 2012 alone [6].

Despite the technical challenges of the immense growth, the success of wireless communication is still far from reaching its full potential. With the advent of the Internet of Things, future trends indicate an exponential increase of wireless devices for machine-to-machine communication. It is envisioned that by 2020 between 50 and 100 billion devices will be connected over the Internet [7]. The visionary applications aim to provide new services for both consumers and businesses. Applications range from smart household utilities, such as fridges that keep track of their content and automatically re-order certain goods, to industry applications such as tracking a commercial product throughout its life-cycle [8]. Additionally, small-scale computers are deployed as wireless sensor networks to enable complex applications to improve factory automation and environment monitoring [9].

Regardless of the particular wireless technology and the application scenario, the main characteristic and challenge of wireless communication is the shared medium that is used to transport messages between the communication endpoints [10]. Due to its broadcast nature, wireless transmissions are potentially received by all network stations in the communication range of the sender. With an unsynchronized medium access, multiple transmissions may be active at the same and thus they interfere with each other. Such interference leads to a high risk that multiple transmissions collide on the receiver side and cannot be properly decoded [1]. Therefore, several challenges result from the competition among the network stations for the medium access [2].

This chapter gives on overview of the basic principles of wireless communication and the corresponding technologies that form the foundation for the work in this dissertation. The

channel assignment challenge is introduced and the chapter closes with an overview of the contributions of the dissertation.

1.1 Evolution of wireless communication

As of today, diverse wireless technologies have been developed and deployed to such a degree that we use many of these often on a daily basis. The different technologies can be classified into different categories. In wireless single-hop networks, clients connect directly to an access point and communicate to other peers only via that particular access point [10]. The most popular examples of these kinds of networks are *cellular networks* and *infrastructure* WLANs [3, 11]. In cellular networks, mobile phones connect to a base station using the licensed frequency spectrum, while in infrastructure WLANs clients connect to an *Access Point* (AP) using the unlicensed frequency spectrum [10].

In contrast, wireless multi-hop networks, are characterized by the absence of infrastructure equipment such as base stations and access points [12]. In order to compensate for the lack of infrastructure, the clients have to perform tasks typically taken care of by the base station such as relaying packets for other nodes over multiple hops towards the final destination. Typical examples of multi-hop networks are *wireless sensor networks* (WSNs) and *wireless mesh networks* (WMNs) [9, 13]. WSNs are usually formed by small-scale low-power devices to enable monitoring applications. WMNs are less constrained regarding the energy consumption and computational power than WSNs and thus are more suitable for data-intensive applications such as providing Internet access in rural areas. It is not uncommon for both network types to have dedicated nodes that provide gateway functionality to connect the networks to the Internet, for example.

The feature of wireless multi-hop networks to be independent of infrastructure equipment makes them suitable in scenarios where a robust telecommunication infrastructure either does not exist or has been disabled, for instance due to a natural catastrophe such as an earthquake [14]. However, the lack of any infrastructure requires multi-hop networks to have a high degree of self-organization in order to achieve a similar reliability as their infrastructure counterpart. Due to relaying the message over multiple hops, the latency and the risk of message loss are higher than in single-hop infrastructure networks. However, an active research community has already solved many of these challenges and is further working on improving the services. The focus of this dissertation is on wireless mesh networks, which are described in more detail next.

1.2 Wireless mesh networks

As stated above, the main characteristic of WMNs is the lack of any infrastructure equipment required for the network operation. In consequence, the mesh routers have to provide the basic services of the base stations such as relaying packets for the other nodes [12]. The typical architecture of WMNs is depicted in Figure 1.2.1. The network architecture

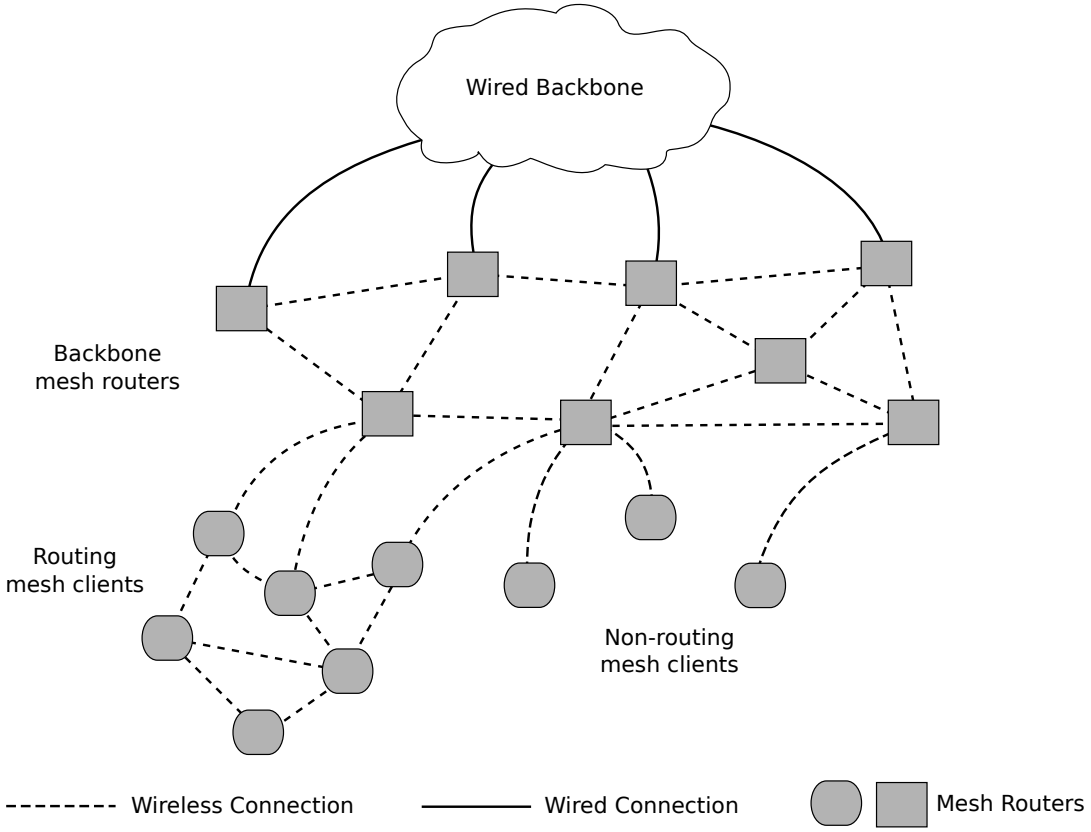


Figure 1.2.1.: Network architecture of wireless mesh networks. The stationary *backbone mesh routers* communicate over the wireless links with each other and with the mesh clients. Backbone mesh routers may have wired connections to another backbone network and can thus function as *gateways* to different networks, such as the Internet. The *mesh clients* may be stationary or mobile and may also function as routers by forwarding traffic for other mesh clients.

comprises three tiers: *gateways*, *backbone mesh routers*, and *mesh clients*. The backbone mesh routers are usually stationary and act as base stations or *access points* for the mesh clients. They communicate over wireless links with each other and with mesh clients. Some of them may have additional wired connections and can thus function as gateways to different networks, such as the Internet. Mesh clients can be stationary or mobile and may function as routers by forwarding traffic for other mesh clients.

The architecture and set-up of wireless multi-hop mesh networks have evolved with technological advances. First, wireless mesh networks were designed as single-radio single-channel networks, in which each network node is equipped with exactly one radio and all nodes operate on the same channel to ensure network connectivity. In single-radio multi-channel networks, nodes change the channel of the radio frequently with changing traffic patterns, which imposes a certain overhead on the performance for the channel switches. With multi-radio multi-channel networks each node is equipped with multiple radios that operate on different channels. This enables the network nodes to communicate simultaneously with multiple neighbors using different radios and channels. The multi-radio

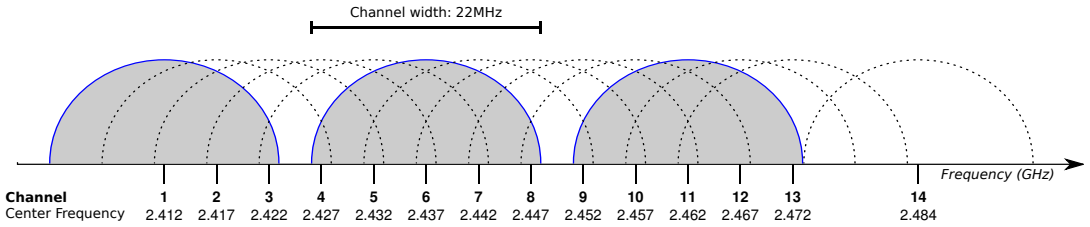


Figure 1.2.2.: Channels of IEEE 802.11b/g. Due to regulatory constraints, channel 14 is only available in Japan. A possible set of three theoretically non-overlapping channels $\{1, 6, 11\}$ is emphasized in gray.

multi-channel set-up is currently the most common type. The reasons for this development are two-fold. First, off-the-shelf *wireless network interface cards* (WNICs) have become significantly cheaper over the last decade [15]. Second, commonly used wireless network technologies, such as IEEE 802.11, provide multiple non-overlapping channels [3].

1.2.1 Technologies for WMNs

Due to the wide availability and low cost of IEEE 802.11 devices, many WMNs rely on this technology. IEEE 802.11b/g supports up to 14 different channels on the unlicensed ISM radio band at 2.4 GHz [3]. The distance of the center frequency of two adjacent channels is 5 MHz with a channel width of 22 MHz as depicted in Figure 1.2.2. The available data rates range from 1 to 11 Mbit/s for IEEE 802.11b and up to 54 Mbit/s for IEEE 802.11g. The IEEE 802.11a standard, intended for the United States, defines 12 non-overlapping channels on the 5 GHz U-NII band. The IEEE 802.11h standard evolved to open the 5 GHz radio band for WLAN devices in Europe. It defines additional restrictions because of the co-existence of military and radar stations that utilize the same frequency band. Available data rates range from 6 to 54 Mbit/s.

The IEEE 802.11n standard [16] offers data rates of up to 600 Mbit/s using *multiple-input and multiple-output* (MIMO) technology [17] with *orthogonal frequency-division multiplexing* (OFDM) [18]. The high data rate is achieved with the spacial diversity due to multiple antennas which allow simultaneous transmission of data streams. IEEE 802.11n operates on the 2.4 GHz as well as on the 5 GHz frequency bands. The IEEE 802.11n standard allows channel-bonding on the physical layer, which means that two non-interfering 20 MHz wide channels can be used simultaneously. This way, the available data rate can be doubled. This feature, also referred to as 40 MHz mode, decreases the number of available non-interfering channels for IEEE 802.11n. An overview of the key features of the different 802.11 standards is given in Table 1.2.1.

The IEEE 802.11s draft [19] addresses WMNs and specifies standard protocols, such as a routing protocol based on AODV [20], for mesh devices in order to create a WMN. Work on the standard started in 2003 and draft 12.0 has been mostly approved in June 2011. A reference implementation for the Linux kernel exists in the `mac80211` module that has been

802.11 standard	Frequency (in GHz)	Modulation	Channel width (in MHz)	Max data rate (in MBit/s)
a	5	OFDM	20	54
b	2.4	DSSS	20	11
g	2.4	OFDM, DSSS	20	54
n	2.4 and 5	OFDM	up to 40	600

Table 1.2.1.: Overview of the different IEEE 802.11 standards.

distributed with the *One Laptop Per Child* (OLPC) initiative [21]. Of interest for channel assignment procedures is that the standard specifies a protocol for channel switching. The procedure is primarily defined to satisfy regulatory requirements. A channel switch may be carried out to avoid interference with a radar signal or to ensure the connectivity of the *Mesh Basic Service Set* (MBSS).

The IEEE 802.16 WiMAX standard also supports multiple non-interfering channels [22]. The standard operates on the unlicensed radio band from 2 to 66 GHz with a flexible channel width, thus theoretically supporting many more non-overlapping channels than IEEE 802.11a/b/g/n. The future of WiMAX is unclear, however, since big manufacturers have ceased the development of hardware after the roll-out of LTE as another 4G mobile technology for IP-based networks [23].

1.2.2 Applications and network deployments

Many fields of commercial and non-commercial applications for WMNs evolved in the last decade. A typical application for WMNs is to provide the last mile infrastructure for Internet Service Providers in dense city areas as well as in rural areas and for university and business networks. Several companies offer hard- and software to set up WMNs for business clients and consumers [24–27]. Self-organized communities, such as the Freifunk community in Berlin [28], MIT Roofnet [29], and Berlin Roofnet [30], set up WMNs to provide Internet access in dense city areas for the participants.

Another use case is emergency scenarios. Disaster management describes a situation after a disaster of natural origin or caused by human beings where the existing communication infrastructure fails, either partially or completely [31]. In this case, a substitution has to be installed in the shortest time possible which should be adequate to support rescue operations. It is obvious that the installed network has to be simple to configure, easy to set up and maintain, and adaptable to a dynamic topology in order to support changes in the number and density of participants [32]. The self-organizing and self-healing characteristics of WMNs make them feasible for these scenarios. Two deployments of such networks in the context of disaster management are the SOSEWIN earthquake early warning system in Istanbul [14] and the Humboldt Wireless Lab-Testbed (HWL) in Berlin [33].

The DES-Testbed at Freie Universität Berlin is a hybrid wireless network consisting of 128 multi-radio mesh routers and a number of wireless sensor nodes in the same order [34].

Research on the DES-Testbed is not restricted to a particular application, as the focus is on development and evaluation of protocols and algorithms on layers 2 - 4 of the network layer stack. The mesh part of the testbed serves as the experimentation environment for all experiments performed in the scope of this dissertation. A detailed description of the DES-Testbed is given in Chapter 3.

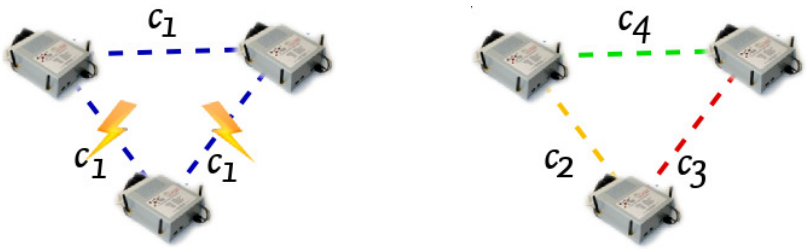
1.3 The channel assignment challenge

Due to the broadcast nature of the shared medium, multiple transmissions in close spatial proximity may interfere with each other. Both the sender and receiver of a wireless transmission may be affected by interference. At the receiver side, simultaneous activity of network interfaces utilizing the same channel in close spatial proximity may result in collisions, thus requiring retransmissions of the message. The sending node may have to defer a message, if the *carrier sense threshold* is exceeded by ongoing transmissions of near-by nodes in IEEE 802.11-based networks [3] which leads to an increase of the latency. It has been shown that interference is one of the major causes for performance degradation in wireless networks [35].

For this reason, protocols and mechanism have been developed on the MAC layer to synchronize the medium access and thus reduce interference effects. One of these approaches is channel assignment. The idea of channel assignment is to minimize the network-wide interference by utilizing non-overlapping (also called non-interfering or orthogonal) channels for otherwise interfering wireless transmissions. This is feasible, since commonly used wireless network technologies, such as IEEE 802.11, provide multiple non-overlapping channels. The key challenge of the problem is how to assign the available channels in a way that interference is minimized and the network performance in regard of the network capacity is maximized while ensuring network connectivity. A resulting channel assignment using different non-overlapping channels can decrease interference but also alter the network topology. When the network interfaces of two neighboring network nodes operate on non-overlapping channels they do not interfere with each other but also the nodes can not communicate directly. Thus, there is a trade-off between channel diversity and network connectivity.

A simple scenario with three nodes forming a single-channel and a multi-channel network is depicted in Figure 1.3.1. To realize the channel assignment of the multi-channel network permanently, two wireless network interfaces for each node are required to establish the links. This leads to the *interface constraint* [36], which states that the number of channels assigned to a node can not be higher than its number of radios for such static channel assignments.

Traditional channel assignment algorithms aim at minimizing the *inter-network* interference, by assigning non-overlapping channels to potentially interfering links in the network. However, since the leading technologies for WMNs, such as IEEE 802.11, operate in the unlicensed frequency spectrum, the dense distribution of private and commercial



(a) Single channel network (b) Multi channel network

Figure 1.3.1.: Channel assignment problem. In (a), channel 1 is assigned to all three links. Thus, simultaneous transmissions may result in interference. By assigning three non-overlapping channels to the links as shown in (b), the links can be activated at the same time without interference.

network deployments of WLANs in urban areas pose a new challenge. These co-located networks compete for the wireless medium operating on the same wireless radio channels, thus decreasing the achievable network performance in terms of throughput and latency. Additionally, other non-IEEE 802.11 devices, such as cordless phones, microwave ovens, and Bluetooth devices, operate on the unlicensed 2.4 and 5 GHz frequency bands as well and can further decrease the network performance [37, 38]. With the realization of the *Internet of Things* (IoT), the number of wireless devices operating in the unlicensed frequency bands is rapidly growing. Therefore, it is an important issue for efficient channel assignment to also address *external interference*. This task is not trivial, since the external networks and devices are not under the control of the network operator and their radio activity is therefore hard to capture. Since their behavior may vary over time, especially considering mobile devices such as smartphones and laptops, it is also hard to predict and model their future behavior. For this reason, many channel assignment algorithms have neglected the impact of external interference.

1.4 Contributions

The contributions of this dissertation comprise the design, implementation, and validation of models and algorithms to enable wireless multi-hop networks to become interference-aware. The overall solution significantly increases the network performance by assigning non-overlapping channels for potentially interfering links and considering the activity of external interferers. The approach has been experimentally validated in a large-scale wireless multi-hop multi-radio testbed. The key contributions are summarized in the following list:

- A measurement-based interference model has been developed, implemented, and validated to identify efficiently and accurately interference relationships between network nodes in real network deployments.
- An external interference-aware channel assignment algorithm that is able to detect

and avoid heavily congested channels has been designed, implemented, and validated in scenarios with high levels of interference.

- A software-based spectrum analyzing solution has been developed, implemented, and validated that enables the detection of external wireless devices and networks.
- A framework to support the development process of channel assignment algorithms has been designed and implemented to allow rapid prototyping of approaches in testbed environments.
- An experimental study of existing channel assignment algorithms in a large-scale wireless testbed delivered insights of the potential performance boost with channel assignment.
- A software-based spectrum analysis solution has been developed, implemented, and validated that enables the detection of external wireless devices and networks.
- A methodology for the performance evaluation of channel assignment algorithms with domain-specific performance metrics and scenarios has been designed, implemented, and used for the evaluation of the algorithms developed in this dissertation.

1.5 Structure

The remainder of the dissertation is structured as follows. Chapter 2 comprises an introduction to the research field of channel assignment in WMNs. Chapter 3 describes the DES-Testbed, the wireless experimentation laboratory used in this dissertation. Chapter 4 presents the applied methodology with the focus on the experiment work flow in testbed environments and the evaluation of channel assignment algorithms. In Chapter 5, the *channel occupancy interference model* (COIM) for the estimation of interference relationships among network nodes is presented. The framework DES-Chan for the development of channel assignment algorithms in testbed environments is presented in Chapter 6. A study on the potential performance gain with channel assignment in a large-scale wireless testbed follows in Chapter 7. An algorithm for external interference-aware channel assignment is presented and evaluated in Chapter 8. A summary and conclusion of the dissertation follows in Chapter 9.

CHAPTER 2

Channel assignment in wireless mesh networks

This chapter introduces the channel assignment problem and the corresponding concepts and terminology in this research field. A literature study of existing distributed algorithms for channel assignment is followed by a discussion regarding their feasibility in large-scale wireless multi-radio networks. The chapter ends with a discussion of existing research frameworks for distributed channel assignment in wireless testbeds.

2.1 Network model

With the *unit disk graph* model [39] a network is represented by an undirected Graph $G = (V, E)$, where V is the set of network nodes and $E \subseteq V \times V$ is the set of wireless links. The graph is undirected, since bidirectional links are required for communication with technologies that require an ACK mechanism on the link layer such as IEEE 802.11 [3]. For simplicity, it is considered that all wireless interfaces have the same transmission radius r_t and interference radius r_i, with $r_i > r_t$. It is usually assumed that $r_i = \alpha \cdot r_t$ with $2 < \alpha < 3$, see [35]. A link exists between two nodes $u, v \in V$ if their distance $d_{u,v}$ is smaller than their transmission radius r_t, more formally $d_{u,v} \leq r_t$. If such a link exists, it is denoted with $l_{u,v}$. Two links $l_{u,v}$ and $l_{x,y}$ interfere with each other if at least one of the following distances $d_{u,x}, d_{u,y}, d_{v,x}, d_{v,y}$ is smaller than the interference radius r_i [40].

It has to be taken into account, that the unit disk graph model comprises all virtual communication links in the network, regardless of a particular interface configuration. To model the channel utilization, in the *network topology* model, a link between $u, v \in V$ exists only if $d_{u,v} \leq r_t$ and at least one interface on each node is tuned to the same channel. A link between the two nodes $u, v \in V$ on channel c in the network model is therefore given with $l^c_{u,v}$. Whereas the unit disk graph model describes all possible links in a WMN, the network topology model only comprises the links that can actually be used for communication.

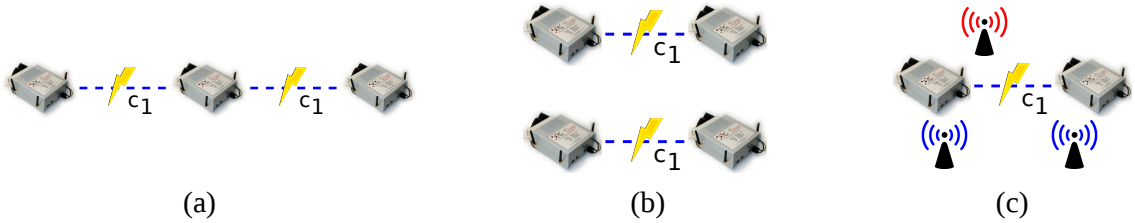


Figure 2.2.1.: Intra-flow, inter-flow, and external interferences. Intra-flow interference may occur when two hops on a path utilize the same channel (a). Inter-flow interference results when two hops of two different flows interfere with each other (b). External interference is exerted from devices which are not under control by the network operator (c).

2.2 Interference

Interference is one of the major causes for performance degradation in wireless networks and may affect both the sender and receiver of a wireless transmission [35]. If messages collide at the receiver side due to simultaneous transmissions on the same channel, retransmissions of the messages are required. On the sender side, the sender may have to defer a message, if the *carrier sense threshold* is exceeded by ongoing transmissions of near-by nodes in IEEE 802.11-based networks [3]. In WMNs, three different sources of interferences can affect the network performance as depicted in Figure 2.2.1. *Intra-flow* interference occurs when successive hops on the path of a single flow utilize overlapping channels. Especially single channel networks are prone to this kind of interference [41]. *Inter-flow* interference results when two links of different flows interfere with each other. As a third source, *external* interference may occur when devices, which are not under control of the network operator, utilize the same frequency band.

Usually, channel assignment algorithms try to reduce inter- and intra-flow interference and leave external interference aside, since it can not be controlled [41, 42]. Still, due to the growing numbers of WMN deployments, especially in urban areas [43], it is likely that external networks co-exist and therefore add to the contention of the wireless medium. These devices are likely to operate in the unlicensed frequency spectrum as well and their radio activity pattern may be very different. They can be distinguished into devices that operate on a fixed frequency (ZigBee [44]), frequency hopping devices (Bluetooth [45]), and broadband interferers (microwave ovens [37]). However, enabling the network with awareness of such external networks and devices without prior knowledge of their location and radio activity pattern is not a trivial task.

2.2.1 Interference modeling

A crucial task for the development of interference-aware solutions is to accurately determine interference relationships between network nodes. The interference relationship for two network nodes u and v describes if transmissions of v may lead to errors in receiving or sending at node u and vice versa. Estimating accurate interference relationships for a

wireless network deployment is difficult due to the complexity of modeling radio signal propagation especially in indoor environments [46]. For this reason, initially simplified models based on heuristics have been proposed for the interference estimation. One of these is the *point interference model*, stating that two links interfere if they share a common endpoint [47]. A more sophisticated model defines for each node (or more precisely each radio when nodes are equipped with multiple radios) a transmission radius r_t and an interference radius r_i, with $r_t < r_i$. A transmission is correctly received at node u, when only one node of all the nodes in the interference range of u transmits at a time. Models based on this assumption usually approximate the m-hop neighborhood of a node u as the *interference set* [48] which consists of all nodes whose transmissions may interfere at u. A typical value is $m = 2$, meaning the interference range of a node is twice its communication range. This model has been first described as *protocol model* [35]. With these simple models, the interference estimation can be calculated without much effort based on the local neighborhood information of a network node. However, several studies have shown, that these simplifications do not hold in reality and the predicted interference relationships were not accurate for indoor network environments [46, 49, 50].

For this reason, *measurement-based* methods have been proposed for interference modeling that promise a higher degree of accuracy [46,51]. In these methods, measurements are carried out in the network to determine the interference relationships between the network nodes. While measurement-based models deliver an higher accuracy in predicting interference relationships, the required effort to perform these measurements is significantly higher [46]. Measurement-based interference models are discussed in detail in Section 5.1.

2.2.2 Conflict graphs

Conflict graphs[1] describe interference relationships between all links in a network and thus quantify the network wide interference level [52]. The conflict graph is constructed based on the communication graph of the network and an interference model. The conflict graph G_C is given with $G_C = (V_C, E_C)$, where V_C corresponds to the edges E of the network graph $G = (V, E)$. An edge in E_C denotes, that the two corresponding edges in V_C interfere with each other according to the utilized interference model.

The conflict graph has been extended to the *multi-conflict graph* (MCG) in [53] in order to model multi-radio nodes. The difference is, that in the MCG, G_M is given with $G_M = (V_M, E_M)$, where V_M is the number of links between wireless interfaces instead of network nodes as in the original conflict graph. The edges E_M are then created in the same way as in the conflict graph, whereas two vertices in V_M are connected with an edge, when the two links between the wireless interfaces interfere with each other. Examples for a conflict and multi-radio conflict graph are depicted in Figure 2.2.2.

[1] Also referred to as *interference graphs* or *contention graphs*.

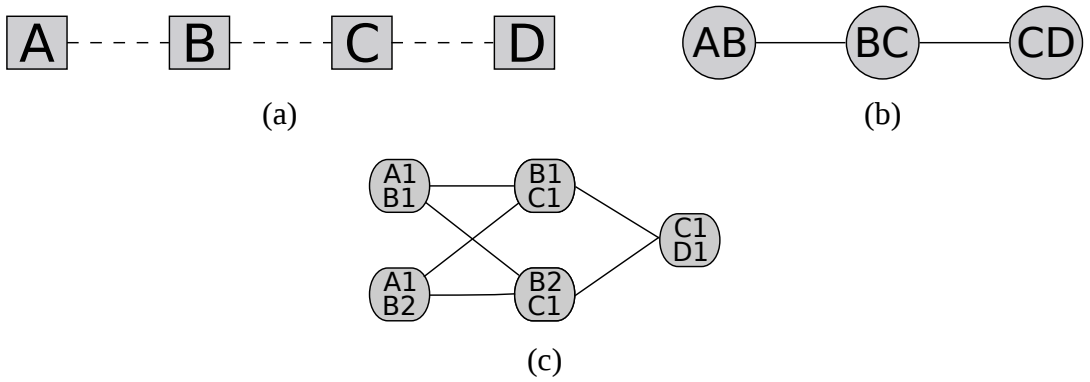


Figure 2.2.2.: Conflict graph and multi-radio conflict graph. (a) shows the network graph $G = (V, E)$ with $V = \{A, B, C, D\}$ and $E = \{(A, B), (B, C), (C, D)\}$. The resulting conflict graph, assuming that only direct neighbors are interferers, is depicted in (b) with the corresponding graph structure $G_C = (V_C, E_C)$ with $V_C = \{AB, BC, CD\}$ and $E_C = \{(AB, BC), (BC, CD)\}$. In (c) the multi-radio graph is depicted when node B is equipped with two radios and all remaining nodes with only one radio.

2.3 Problem formulation of channel assignment

The idea of channel assignment is to minimize the network-wide interference by utilizing non-overlapping channels for otherwise interfering wireless transmissions. By exploiting the benefits of orthogonal channels for otherwise interfering transmissions, channel assignment aims at improving the network performance, especially in regard to the network capacity. The key challenge is how to assign the available channels in a way, that interference is minimized while ensuring the network connectivity. A resulting channel assignment using different non-overlapping channels can decrease interference but also alter the network topology. When the network interfaces of two neighboring network nodes operate on non-overlapping channels, they do not interfere with each other but also the nodes can not communicate directly. Thus, there is a trade-off between the channel diversity and the network connectivity. For this reason, most algorithms introduce mechanisms to preserve the network topology throughout the channel assignment procedure, for instance by tuning one interface per node permanently to a common network-wide global channel [48]. The problem of channel assignment has been reduced to variations of the graph coloring problem and has thus proven to be NP-hard [36,40,54,55]. Thus, most proposed distributed channel assignment algorithms build on greedy methods in order to locally minimize the interference levels.

Several challenges for efficient distributed channel assignment algorithms have been identified. *Channel oscillation* describes the situation in which two links periodically switch channels because the channel switch for the first link leads to sub-optimal channel assignment on the second link and vice versa [41]. Another undesirable phenomenon is the *ripple effect* which describes the situation in which one channel switch is followed by several subsequent channel switches throughout the network [56]. This happens when the channel of

the first hop of an $\mathfrak{m}$-hop path is changed, and in consequence the channel assignment is not optimal anymore on the second hop. Therefore, the channel is switched on the second hop as well, and the situation repeats for all $\mathfrak{m}$ hops on the path. The described phenomena may prevent the stabilization and convergence of distributed channel assignment algorithms. Therefore, it is important to develop appropriate counter-measures, such as the synchronization of channel switches. Some algorithms also introduce restrictions regarding the channel assignment choices, such that each channel-link combination may only be assigned once throughout the channel assignment procedure [36, 40]. While this helps guaranteeing the convergence and stabilization of the algorithm, it may lead to sub-optimal channel assignment and render the algorithm unadaptive to topology changes.

2.4 Classification of channel assignment algorithms

Over the last decade, many channel assignment algorithms have been developed for different scenarios [41, 57, 58]. This resulted in a wide range of different approaches tailored to specific network technologies, network architectures, and application scenarios. Because of the variety, the algorithms can be classified according to the following key properties.

First, channel assignment algorithms can be considered *centralized* or *distributed* [41]. Centralized channel assignment approaches rely on a central control instance that calculates the channel assignment for the whole network. This entity is often referred to as *channel assignment server* (CAS) [53]. The CAS gathers network topology information, calculates the channel assignment based on the global network view, and notifies the network nodes about the result to adjust their channel assignment accordingly. In distributed channel assignment approaches, the algorithm runs on every network node and executes channel assignment decisions taking only local information into account. The communication overhead for a CAS does not exist in distributed approaches. Still, communication among the nodes is necessary to exchange local information and to notify neighbors of changes in channel assignment. Distributed algorithms are considered to be less prone to node failures because they do not rely on a CAS which may constitute a *single point of failure* [41]. Also, they are usually more adaptive to a dynamic network topology in regard to node mobility and node failures because the corresponding topology changes can be handled locally. Still, distributed algorithms lack the advantage of using a global network view for the channel assignment calculations. This may lead to suboptimal results considering the network-wide channel assignment.

Distributed approaches must also solve challenges concerning the retrieval of the channel usage in their neighborhood and the propagation of channel switches. Usually neighborhood information is exchanged periodically, which also allows to adapt to network topology changes. In order to ensure that channel switches are noticed and do not result in dead interfaces or channel oscillation, channel switches are usually propagated in the neighborhood with simple handshake protocol mechanisms [41].

Another classification considers the frequency of channel switches on a network node. In

literature, channel assignment approaches are classified as *dynamic* or *static*. In *dynamic* approaches, channel switches may occur frequently, in the extreme for every subsequent packet a different channel is chosen [59]. The limiting factor for dynamic algorithms is the relative long channel switching time with commodity IEEE 802.11 hardware, which is in the order of milli seconds [60]. *Static* approaches in contrast, switch the interfaces to a particular channel for a longer period, usually in the order of minutes or hours [48]. *Hybrid* approaches combine both methods.

2.5 Related work: Distributed channel assignment algorithms

Many algorithms for channel assignment in multi-channel multi-radio networks have been proposed in the last years. Therefore, only a selection of approaches is discussed in the following. The focus of this related work study is on distributed and external interference-aware algorithms, which are closest to the research contribution of this dissertation. For a more complete list, we refer to the surveys of algorithms in [41,57,61].

The algorithms described in this section make several common assumptions, for example that a number of non-overlapping channels C are available and that the number of interfaces K on every node is smaller than C ($C \gg K$). Usually, the algorithms consider the mesh network as a stationary backbone, meaning that the mesh routers are immobile, although some algorithms consider node failures and low mobility, which makes them adaptive to topology changes. Also, all surveyed algorithms consider only bi-directional links because of the ACK-mechanism of IEEE 802.11 unicast transmissions. After a description of the different approaches, we introduce a classification for a conclusive discussion.

2.5.1 Distributed Greedy Algorithm - DGA

The *distributed greedy algorithm* (DGA) assigns channels to network nodes, or more precisely, to network interfaces [48,62]. Intuitively, each node tries to minimize the interference by assigning the least used channels in its *interference set*. The interference set of a node n consists of all nodes and their channel assignment whose transmissions affect sending and receiving at node n. To preserve the network connectivity, one interface on every node is operating on a global common channel and is not used for channel assignment. For any additional interfaces (referred to as *switchable* interfaces), the channel is determined based on an *interference cost function* f_I as given in the following equation.

$$f_I(f_1, f_2, \alpha) = max(0, \alpha - |f_1 - f_2|) \tag{2.1}$$

The spectral overlap of two channels is used to determine the level of interference in the interference set. The cost function takes the center frequencies f_1 and f_2 of two radio channels and the additional parameter α into account. α denotes the minimum frequency difference of orthogonal channels. A value of $f_I = 0$ means that the channels are orthogonal and thus do not interfere with each other. In each iteration, the algorithm considers exactly

one switchable interface and calculates the interference cost for all available channels K with the channel assignment of the interference set. The result is the channel c_{best} with the lowest overall interference cost. If c_{best} is different than the currently assigned channel for the interface in question, a channel switching procedure based on a three-way handshake including all nodes of the interference set is initiated. As an additional constraint, only channels can be assigned that are used by at least one neighboring node, in order to avoid isolated interfaces.

The advantage of the algorithm is its simple greedy nature and the proven convergence [62]. The interference cost function is simple to calculate but only relies on the spectral distance of the available channels. Spatial distance between nodes, obstacles, and external interferences which have an impact on the interference and link quality in real networks are not considered.

2.5.2 Minimum interference channel assignment - MICA

The *minimum interference channel assignment* (MICA) assigns channels to links instead of interfaces and is therefore topology preserving [36]. In other words, all links in the single channel network, also exist in the multi channel network after the channel assignment procedure. This way, the approach is independent of the overlaying routing algorithm. The authors formulate the channel assignment problem as minimizing the *total network interference* by minimizing the number of edges in a conflict graph [52]. The problem is reduced to the Max K-cut problem, thus proving that it is NP-hard. The proposed distributed algorithm is based on a greedy approximation algorithm for the Max K-cut problem given in [63].

At the network initialization, all links are assigned to the same channel. After retrieving the local communication graph of the m-hop topology (with $m \geq 1$) and constructing the conflict graph V_c, each nodes iterates over all link-channel combinations (u, c), identifying the combination which results in the largest decrease of interference in the local neighborhood. The largest decrease is achieved with the combination (u, c) that removes most edges in the local conflict graph V_c compared to all other combinations. Thereby, the interface constraint has to be respected, which means that no more channels can be assigned to a node than it has interfaces. Before the network node n selects the link-channel combination (u, c), the following criteria have to be met.

- Node n is incident to the link u
- Node n *owns* link u, meaning that its ID is higher than the ID of the other node incident to u
- Applying channel c does not violate the interface constraint on both nodes
- The combination (u, c) has not been tried before at node n

- The combination (u, c) results in the highest decrease of interference compared to all other combinations

The algorithm terminates, if no further (u, c) combination can be found that matches all criteria. Once a combination (u, c) is found, node n initiates a channel switching procedure with the corresponding neighbor of the link in question.

As the main advantage due to the link-based channel assignment, the algorithm does not alter the network topology. Thus, routing algorithms can operate independently on top of the channel assignment. The main weakness of the algorithm is the restriction that each link and channel combination can only be changed once, thus the algorithm is not adaptive to topology changes.

2.5.3 Sridhar - SRI09

Sridhar et al present an extension of the previously discussed MICA algorithm that additionally takes the expected traffic-load on the nodes into account [40]. The algorithm differs in that a link is now owned by the node whose expected traffic load is highest, compared to the ID-based approached in the original MICA.

Due to the strong similarity, the same advantages and weaknesses account for this algorithm as to the previous one. The main difference is the extension with a traffic load matrix of expected traffic for every link, which enables this approach to perform load-aware channel assignment.

2.5.4 Breadth-first search channel assignment - BFS-CA

The first algorithm for channel assignment that considers external interference has been proposed in [53]. The centralized algorithm incorporates the external interference by periodically assessing the interference levels of the available channels. This is done by setting a wireless interface into *monitor mode*, in which frames of co-located IEEE 802.11 stations can be received that pass the *frame check sequence* (FCS). This way, co-located IEEE 802.11 interfaces in the communication range can be detected. The congestion of the channel is measured by analyzing the captured packets for the particular channel. The results are passed to a CAS which then calculates the network wide channel assignment and sends the decisions back to the mesh nodes.

Once the topology and channel condition information is collected at the CAS, the channel assignment is calculated based on a *breadth-first search* (BFS) algorithm originating at the gateway. Thus, links closer to the gateway are prioritized. After sorting the links according to the distance to the gateway, the highest ranked channels are assigned to them taking the channel conditions into account.

The algorithm has been the first one to introduce a measurement-based approach capturing the channel conditions in the channel assignment procedure. However, with this method only IEEE 802.11 radios are captured. Also, *remote interferers*, which reside outside the communication range but inside the interference range, are not captured because

their transmissions would not pass the FCS. The same method to detect external interference with an interface in monitor mode is also used in [64]. A hybrid channel assignment algorithm is then extended that considers the channel conditions.

2.5.5 Net-X

A MAC-layer protocol as a joint solution of channel assignment and routing is proposed in [65, 66]. The set of network interfaces on each node are divided into *fixed* interfaces, which stay on a fixed channel, and *switchable* interfaces, which can be switched dynamically to particular channels. If a node wants to communicate with a neighbor, it tunes one of the switchable interfaces to the channel of a fixed interface of the receiving node. Thus, the algorithm qualifies as an hybrid approach.

The crucial part of this approach is the way how channels are assigned to the fixed interfaces. The authors present two different algorithms for this task. For a simple solution, a function may be used which calculates the channel for a fixed interface based on the node ID. As an alternative, neighborhood information is considered for the channel selection. For this, HELLO-packets with the assignment of fixed radios are exchanged periodically, which allows each node to learn about the channel assignment in the 2-hop neighborhood. Each node selects the least used channel for its fixed radio. While the former approach is only feasible in a static and known network topology, the latter introduces more overhead but can adapt to topology changes.

Additionally, the challenge of broadcasting in a multi-channel network is addressed. The presented solution is to send a broadcast packet once for each channel. The algorithm uses one packet queue for each channel. By considering the queue size of each channel and defining minimum and maximum periods for a switchable interface to stay on one channel, the number of channel switches can be reduced. The NET-X framework implements this approach in a testbed environment which is presented in Section 2.7.2.

The advantages of this approach are that no default channel has to be used to ensure the network connectivity and that fairness is addressed with the queue management. A weakness constitutes the high implementation effort to ensure an efficient dynamic channel assignment. Changes in the Linux kernel have been necessary to reduce the channel switching time and implement channel-based packet queues.

2.5.6 Urban-X

Urban-X provides a cross layer architecture for multi-radio wireless mesh networks tailored to urban scenarios with co-located wireless devices operating in the unlicensed frequency bands [67,68]. It comprises a channel assignment algorithm, a spectrum sensing component to assess external interference, and an extended AODV multi-path routing protocol.

The spectrum sensing component is based on an energy detection scheme to measure the channel busy time. Thus, the channel load can be estimated. The channel assignment algorithm is an extension of the algorithm initially proposed with NET-X [65]. On

each network node, one interface is set to a *fixed* channel for receiving and further *switchable* interfaces can be switched dynamically to particular channels. If a node wants to communicate with a neighbor, it tunes one of the switchable interfaces to a channel of a fixed interface of the receiving node. The channel assignment algorithm for the fixed interfaces incorporates the estimated load per channel and the current internal network channel assignment.

The algorithm has the advantage to consider external interference and the spectrum sensing component is promising for the task to estimate the channel load. However, Urban-X has been realized so far as a simulation framework in which modeling the signal propagation in complex environments is limited.

2.5.7 Skeleton assisted partition free channel assignment - SAFE

The *skeleton assisted partition free channel assignment* (SAFE) algorithm preserves a minimal spanning tree (MST) of the network topology [69]. Thus, the network connectivity is ensured, but not all links of the single channel network may exist after the channel assignment procedure. The algorithm comprises two components. A random channel assignment is applied if $K < 2 \cdot C$, where C is the number of wireless network cards on every node and K the number of non-overlapping channels. Due to the pigeonhole principle, two nodes will share a common channel although they are assigned randomly, thus preserving the network connectivity. The second component of the algorithm introduces the condition that all edges of a MST, the *skeleton*, of the network have to be preserved when $K \geq 2 \cdot C$. For this, every node randomly chooses a channel set with $C - 1$ channels, leaving one interface unassigned. The node broadcasts its chosen channel set, and if links to all skeleton neighbors are already established it assigns a random channel to the unassigned interface. Otherwise it tries to establish links with the not connected skeleton neighbors by assigning a channel which is in the channel set of all skeleton neighbors. If there is such a channel, it is assigned to the interface, if not, a global common channel is used for these links.

The advantage of this algorithm is that it ensures the connectivity of the network using a MST of the virtual network links. Still, the algorithm is prone to node failures since this loss of a skeleton edge may partition the network.

2.5.8 Neighborhood nodes collaboration to support QoS - NNCQ

The *neighborhood nodes collaboration to support QoS* (NNCQ) algorithm is a joint approach for routing and channel assignment [70]. At the network initialization, all nodes possess a connectivity matrix containing all available network links. During the network operation, the packet loss rate of a particular link is used as channel switching metric, which is calculated periodically by monitoring sent and received packets.

Based on this metric, the algorithm consists of two phases:

1. *Monitoring phase*: The nodes monitor the packet loss rate for all adjacent links.

2. *Channel switching phase*: If the packet loss rate for a link reaches a certain threshold, the channel switching phase is executed. In this phase, the sending node searches a different node-disjoint path using the connectivity matrix. If a new path is found, the sender initiates a channel switch with the next hop on the path. The least used channel for the link is chosen.

After the successful channel switch, multiple routes for the same destination are available. Therefore, a route selection process has to be executed. A source routing is used with NNCQ and either the best route is chosen for all transmission or the routes are used round-robin like. The latter method may result in frequent channel switches.

The advantage of this approach is the consideration of the packet loss rate which also incorporates external interference and allows nodes to quickly react to link quality changes. As a weakness, each node relies on a global connectivity matrix that is never updated and thus renders the approach unable to cope with network topology changes.

2.6 Discussion

2.6.1 Classification Keys

The following classification keys are used to characterize the presented channel assignment algorithms.

- Distributed algorithm (DA): Defines if the algorithm is of distributed nature.
- Channel Switching Frequency (CSF): Defines the frequency of channel switching. In *dynamic* or *fast channel switching* approaches, channel switches may occur frequently, up to for every subsequent packet. In *static* or *slow channel switching* approaches interfaces are switched to a particular channel for a longer period. *Hybrid* approaches combine both methods.
- External Interference (EI): Defines if external interference is considered by the algorithm. This may be addressed directly by detecting sources of external interference or indirectly by measuring the packet loss rate of a particular link.
- Link Connectivity Preserved (LCP): Defines if all virtual links of the network are preserved after channel assignment.
- Conflict Graph Minimization (CGM): Defines if the problem is formulated such that the number of edges in the conflict graph shall be minimized.
- Interference Model (IM): Defines which interference model is used for the interference estimation.
- Failure / Mobility (FM): Defines the degree of adaptivity of the algorithm. Node failures and node mobility lead to network topology changes, which result in nodes joining and leaving the neighborhood of other nodes.

- Fairness (FA): Defines if the approach considers fairness in regard to network resources.
- Testbed Evaluation (TE): Defines if the approach was implemented and evaluated in a testbed environment.
- Traffic Load (TL): Defines if the expected traffic load is considered in the algorithm.
- Channel Oscillation (CO): Defines if the channel oscillation problem is addressed by the algorithm
- Routing Metric (RM): Defines which routing metric is used for the evaluation to exploit channel diverse paths.

The characteristics of each algorithm in regard to the classification keys are summarized in Table 2.6.1. The following discussion compares the different channel assignment algorithms and investigates trends for future algorithms.

2.6.2 Summary

All of the presented algorithms are of distributed nature apart from BFS-CA [53]. In this centralized algorithm, a dedicated channel assignment server (CAS) calculates the network-wide channel assignment. The algorithm is listed, since it was the first one to consider external interference by using spectrum sensing to assess the channel conditions.

Of the surveyed approaches, all but NNCQ [70], NET-X [66], and URBAN-X [67] are static, meaning that channels are assigned to interfaces for a longer period of time. One reason for this is the relative long channel switching time, which is in the order of milliseconds for current IEEE 802.11 hardware [60]. Also the implementation of dynamic channel assignment is more complex. For instance, NET-X [66] and URBAN-X [67] require changes to the Linux kernel and the drivers of the wireless interface in order to reduce the channel switching time [65]. New hardware is likely to reduce the channel switching time, thus making dynamic schemes more attractive for future approaches.

Most of the discussed algorithms try to reduce inter- and intra-flow interference and leave external interference aside, since it can not be controlled. However, with the focus shifting to urban scenarios with a dense distribution of co-located wireless devices and networks, external interference has gained more attention recently. Of the discussed algorithms, BFS-CA [53], Urban-X [67], and NNCQ [70] consider external interference in the channel assignment. While BFS-CA [53] and Urban-X [67] use spectrum sensing to assess the channel conditions, NNCQ [70] indirectly adapts to external interference by periodically monitoring the packet loss rate for a given link.

Algorithm	DA	CSF	EI	LCP	IM	CGM	FM	FA	TE	TL	CO	RM
DGA	•	static	–	•	3-hop NH	–	–	–	•	–	•	WCETT
MICA	•	static	–	•	m-hop NH	•	–	–	•	–	•	–
SRI09	•	static	–	•	Fixed range	•	–	–	–	•	•	–
BFS-CA	–	static	Spectrum sensing	•	2-hop NH	•	–	–	•	•	•	WCETT
Net-X	•	hybrid	–	•	2-hop NH	–	–	•	•	–	•	MCR
Urban-X	•	hybrid	Spectrum Sensing	•	2-hop NH	–	–	•	–	–	•	Custom
SAFE	•	static	–	–	2-hop NH	–	•	•	–	–	–	–
NNCQ	•	hybrid	Packet Loss Rate	•	Packet loss	–	–	–	–	•	–	–

Table 2.6.1.: Overview of channel assignment algorithms with the corresponding classification keys: DA: Distributed algorithm, CFS: Channel Switching Frequency, EI: External Interference, LCP: Link Connectivity Preserved, IM: Interference Model, CGM: Conflict Graph Minimization, FM: Failure / Mobility, FA: Fairness, TE: Testbed Evaluation, TL: Traffic Load, CO: Channel Oscillation, RM: Routing Metric.

All surveyed approaches preserve the network connectivity in order to avoid network partitions caused by channel assignment. All approaches apart from SAFE [69] also preserve the link-based connectivity. This means that all virtual network links between nodes are preserved. Operating one interface per node on a common global channel to preserve the link-based connectivity is done in DGA [48] and BFS-CA [53].

Most of the surveyed approaches use simple distance-based interference models and define the interference set as the $\mathfrak{m}$-hop neighborhood, with $2 \leq \mathfrak{m} \leq 3$. BFS-CA [53] states that the used 2-hop model is exchangeable with a more accurate, measurement-based model but the effort for measurements is not in the scope of the algorithm study.

With MICA [36], SRI09 [40], BFS-CA [53], several approaches formulate the channel assignment problem as minimizing the number of the edges in the conflict graph or multi conflict graph. Of these algorithms, only MICA [36] has been evaluated with the *fractional network interference* (FNI) metric as described in Section 4.2.

Adaptivity to topology changes caused by node failures and mobility is not of a high concern of the presented approaches. MICA [36] and SRI09 [40] are not adaptive because they prevent the re-assignment of a channel-interface combination in order to ensure the convergence of the algorithm. Adaptivity is addressed by running the algorithm periodically in DGA [48] and SAFE [69]. One reason why adaptivity is not considered an important feature is the assumed network architecture which is a stationary mesh backbone. Still, adaptivity becomes more important with the trend to more volatile, large-scale deployments with an increasing number of mobile nodes and possible node failures.

The algorithms BFS-CA [53], Net-X [66], Urban-X [67], and SAFE [69] consider fairness among the nodes for the channel assignment decisions. The fair distribution of network resources is of higher concern in gateway-oriented network architectures such as [56, 71], in which traffic patterns are more predictable and are usually limited to flows from mesh router to gateway and vice versa. In the presented approaches the expected traffic load is only considered in SRI09 [40] and NNCQ [70]. The reason for this is that most approaches are targeted at peer-to-peer scenarios.

An implementation candidate for a testbed environment has been evaluated for DGA [48], MICA [36], BFS-CA [53], and Net-X [66]. In the field of channel assignment, a testbed evaluation is of high value due to the complex modeling of signal propagation and the resulting interference effects.

Most approaches prevent channel oscillation by using a three-way handshake to announce channel changes as in DGA [48], MICA [36], SRI09 [40], and NNCQ [70]. Channel oscillation can not occur with centralized algorithms, such as BFS-CA [53].

Several algorithms have been evaluated with interference-aware routing metrics. DGA [48] and BFS-CA [53] use WCETT [72], while Net-X uses MCR [66]. In the scope of Urban-X [67], a custom routing metric has been designed, incorporating the packet transmission delay, packet error rate, and the channel switching time.

Based on the discussion of the algorithms and classification keys, we try to predict the trends for future research in channel assignment. With the exploding number of wireless

devices, channel assignment algorithm will have to consider external interference to avoid an significant impact on the network performance [37,38]. Especially since many applications of WMNs, such as providing Internet access to mobile clients, are envisioned mainly for urban scenarios. With the trend towards large-scale network deployments in regard to the number of nodes, the need for the algorithms to be adaptive towards node mobility, node failure, and changes in the channel conditions becomes more important. Finally, due to the huge effort to obtain measurement-based interference estimations, all algorithms rely on simple distance-based interference models. The recent development of more efficient measurement-based approaches [50, 73] combined with the results of studies showing the inaccuracy of the simple models [46, 49], will allow channel assignment algorithms to profit of the more accurate interference estimations.

2.7 Related work: Frameworks for distributed channel assignment

The implementation process of channel assignment algorithms in testbed environments yields many challenges and pitfalls, since common operating systems are not designed to support channel assignment out of the box. Thus, the programmer has to deal with operating system specifics, drivers for the wireless interfaces and the capabilities and limitations of the particular hardware. This is also reflected in the results of the survey of the previous section, in which only about half of the algorithms have been evaluated in a testbed environment.

A development framework for channel assignment algorithms can simplify the implementation effort by introducing an *abstraction layer* for common operating system specific tasks, such as channel switches and the configuration of the wireless interfaces. Furthermore, the analysis of several diverse distributed algorithms has shown that despite their different strategies for channel assignment, a couple of common key services are required for almost all distributed algorithms. This includes a neighbor discovery service, node to node message exchange for channel switch synchronization, and interference models for the network-wide interference estimation and for creating conflict graphs. By providing a *service library* of such commonly used services, a framework for channel assignment can significantly speed up the development process. This is especially helpful, if several strategies are implemented to compare their performance. As next, we present frameworks that have been created to support researchers in the development process of channel assignment algorithms.

2.7.1 FreeMac

With the release of the MadWiFi driver and the OpenHAL [74], an open source implementation of the binary *hardware abstraction layer* (HAL) distributed by Atheros, several frameworks emerged to support research on MAC protocols for commodity IEEE 802.11 hardware. FreeMAC [75], based on its predecessor MadMAC [76], is such a framework

that provides an API to simplify multi-channel MAC protocol development.

The channel switching time is a crucial issue for the performance of dynamic channel assignment algorithms, and therefore, FreeMAC has altered the channel switching procedure to make it faster. Any neighbor discovery operations that usually take place after a channel switch have been removed. The achieved channel switching time, including a hardware reset of the wireless network interface and operating system specific operations, was measured on a Linux-based test system with about 4 ms [75].

FreeMAC can function as the foundation of dynamic channel assignment algorithms that require fast channel switching. However, the framework does not provide any other services for channel assignment algorithms, such as neighbor discovery or communication among network nodes.

2.7.2 NET-X

The NET-X framework was created to implement the fast channel switching algorithm described in [66] for a wireless testbed environment based on the 2.4 Linux kernel [65, 77]. The algorithm divides the available wireless network interfaces on each node into *fixed* interfaces, which stay on a single channel for receiving, and *switchable* interfaces for sending, which can be dynamically switched to different channels depending on the receiver. The research focus of the NET-X framework is on algorithms for the channel assignment of the fixed interfaces.

The implementation of the NET-X algorithm requires several changes to the Linux network stack and to the driver of the wireless network interfaces which is supported by the framework. Among them are per-channel queues, in order to store messages temporarily that are to be transmitted on a different channel than the switchable interface is currently operating on and the reduction of the channel switching time. The architecture of NET-X comprises the *channel abstraction layer* (CAL), the *kernel multi-channel routing* (KMCR), and the *userspace daemon* as shown in Figure 2.7.1. The CAL comprises broadcast and unicast components, queuing and scheduling mechanisms for the channel queues, and implements the configuration of the wireless network interfaces through the userspace daemon. The KMCR provides the necessary functionality to support reactive routing protocols, such as packet buffering during the route discovery procedure. The userspace daemon comprises a multi-channel routing protocol and the interface handling. An abstraction layer for interface handling is provided with a set of functions to control the *interface capabilities*. These comprise the radio parameters that affect the network performance, such as the utilized channel and transmission power.

Experiments with the NET-X framework have been performed to evaluate different strategies for the channel assignment of the fixed interfaces [78] and for QoS-provisioning. Extensions have been developed for queue management [79]. In summary, the services provided by NET-X are directly derived from the requirements of the implemented algorithm [66]. The framework is not intended to function as a foundation to develop different

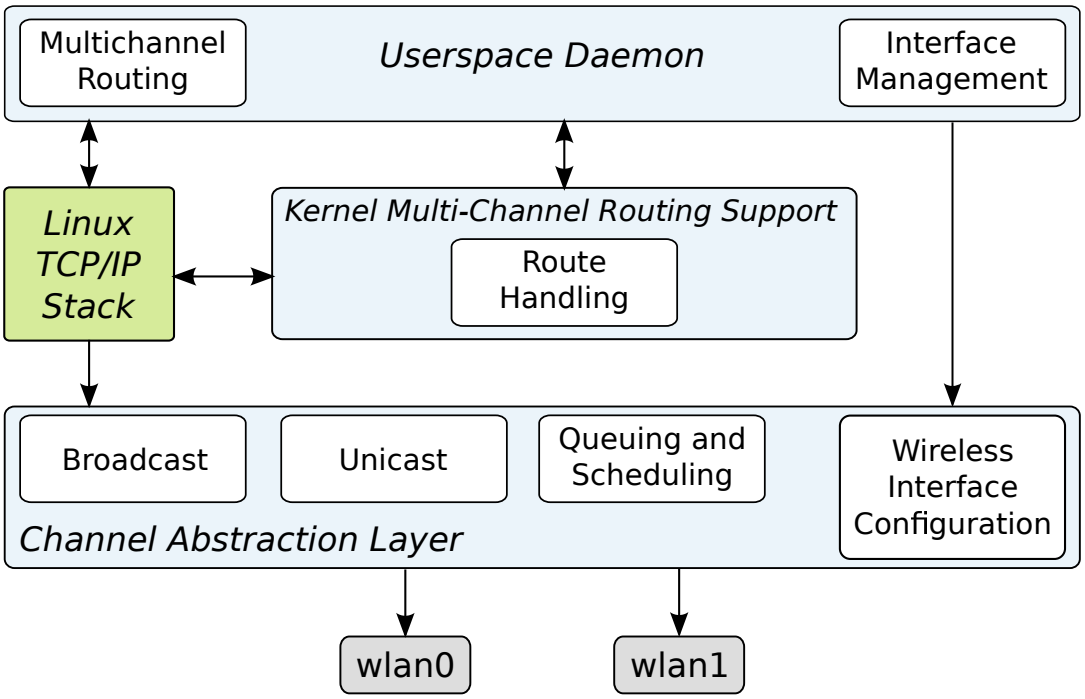


Figure 2.7.1.: Architecture of NET-X. Displayed are the three components of NET-X the *channel abstraction layer* (CAL), the *kernel multi-channel routing* (KMCR), and the *userspace daemon* and their interconnection.

distributed channel assignment algorithms, the focus is on algorithms to assign channels to the fixed interfaces. Data structures for network and conflict graphs are not provided, also a topology monitoring functionality has not been in the scope of the framework.

CHAPTER 3

DES-Testbed: Wireless experimentation laboratory

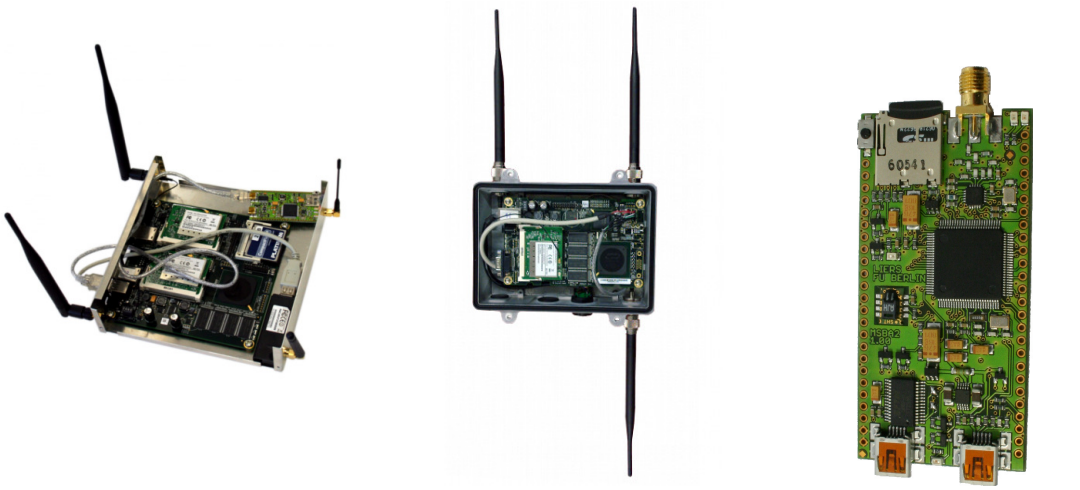
The *Distributed Embedded Systems Testbed* (DES-Testbed) is a hybrid wireless network located on the campus of Freie Universität Berlin. It is hybrid in a way, that all DES-Nodes comprise a wireless mesh router equipped with multiple IEEE 802.11a/b/g radios and an MSB-A2 sensor node [80]. The deployment of the testbed started in 2008 and since then, the testbed has been used for experimentation in several European (OPNEX [81], WISEBED [82]) and national funded projects (G-Mesh-Lab [83]). All experiments described in this dissertation have been performed on the DES-Testbed.

The chapter starts with a description of the network architecture and the hard- and software components of the DES-Testbed. The experimentation methodology presents the workflow for testbed experiments and the corresponding tools to assist researchers in the studies. The chapter closes with experimental results concerning the network topology and the characteristics of the IEEE 802.11 radio channels.

3.1 Architecture

The DES-Testbed comprises about 128 indoor and outdoor DES-Nodes. The hybrid DES-Nodes consist of a *mesh router* and a *sensor node* (see Figure 3.1.1) in the same enclosure, thus forming an overlapping WMN and WSN. The DES-Nodes are deployed in an irregular topology across several buildings on the computer science campus as depicted in Figure 3.1.2. Besides the DES-Testbed, several in-parallel IEEE 802.11 networks exist in order to provide Internet access to students and staff members on the campus. These networks are not under our control and thus contribute to external interference. We treat this as a condition that is also likely to be expected in a real world scenario. A study on the levels of external interference generated by these co-located networks is presented in Section 8.2.4.

Each DES-Node is equipped with three IEEE 802.11 radios. One of the interfaces is a



(a) Indoor DES-Node (b) Outdoor DES-Node (c) MSB-A2 sensor node

Figure 3.1.1.: DES-Nodes of the DES-Testbed. The indoor DES-Node (version 2) with a custom enclosure is shown in (a), the outdoor DES-Node in (b). An MSB-A2 sensor node shown in (c) is attached via USB to each DES-Node and resides in the same enclosure.

Ralink RT2501 USB stick and the other two are Mini PCI cards with an Atheros AR5413 chipset. The cards use the *rt73usb* [84] and *ath5k* [85] drivers, which are part of the Linux kernel. While the Ralink WNICs are IEEE 802.11b/g devices operating in the 2.4 GHz band, the Atheros WNICs additionally support the IEEE 802.11a standard on 5 GHz. The MSB-A2 sensor nodes of the DES-Testbed are equipped with a CC1100 transceiver operating in the ISM bands at 863 to 870 MHz. The radio channels utilized by the MSB-A2 are therefore orthogonal to the frequency band used by the IEEE 802.11 WNICs.

In theory, not all 14 channels in the 2.4 GHz and 24 channels in the 5 GHz band can be legally used for the ad-hoc mode in Germany due to regulatory requirements. The *regulatory domains* are defined by the European Telecommunications Standards Institute (ETSI) [86] for Europe, the Federal Communications Commission (FCC) [87] for the USA, and the Telecom Engineering Center (TELEC) [88] for Japan. However, since the DES-Testbed is an experimental environment, we removed all regulatory constraints. A static regulatory domain database for the Linux kernel allows us to remove all restrictions. Additionally, we discovered a hard-coded limitation for the ad-hoc mode in the upper 5 GHz band for the `ath5k` driver for the Linux kernel version 2.6.34 which had to be removed as well. As a result, all available channels on both frequency bands can be used for experimentation purposes on the DES-Testbed. Further details of the architecture of the DES-Testbed are available in the technical reports [89–91].

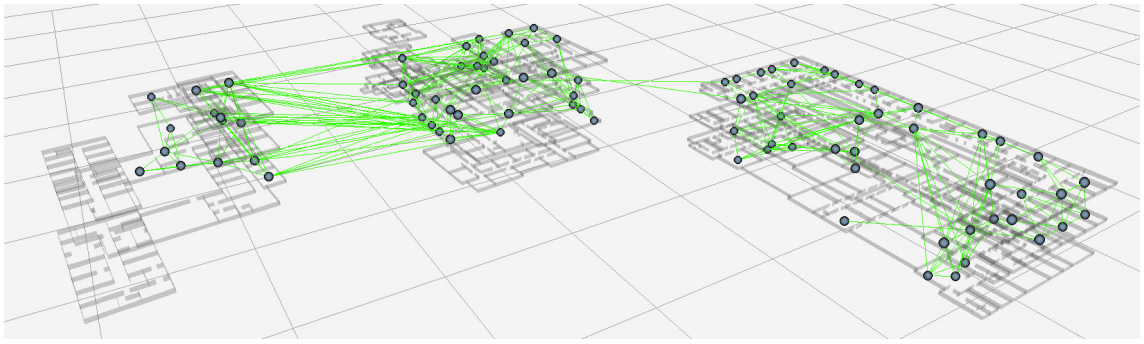


Figure 3.1.2.: Snapshot of the DES-Testbed topology. The DES-Nodes are distributed over several buildings on the computer science campus.

3.2 Methodology

An important design goal of the DES-Testbed is to support the experimenter throughout all steps of the scientific workflow. The first crucial challenge is to provide convenient and reliable means to access the testbed and the network nodes for experimentation. The most open way is to enable remote experimentation, which means that scientists can access the testbed and run experiments remotely over the Internet. Additionally, an experimental environment may provide appropriate tools that guide the researcher throughout the experimental process and help to avoid pitfalls. Before describing the available tools for the DES-Testbed, we first define the task of experimentation and the experiment workflow for network experiments in general.

3.2.1 Experimentation

Independent of particular experiment environments and research subjects, requirements exist which have to be fulfilled by all researchers and their experiments to ensure meaningful results. We consider the following requirements as the most important ones each experiment should meet:

- *Unbiased:* An experiment should be as general as possible and should not be modified to fit a particular experiment environment or setting.
- *Rigorous:* An experiment should focus only on the crucial matter it wants to investigate to reduce side effects.
- *Repeatable:* An experiment should be executed as often as desired to obtain statistically sound results.
- *Reproducible:* Results from different executions of an experiment in the same environment should deviate only slightly.

Whereas some of these requirements depend fully on the scientist, such as the invention and design of experiments [92, 93], experiment environments are capable of supporting

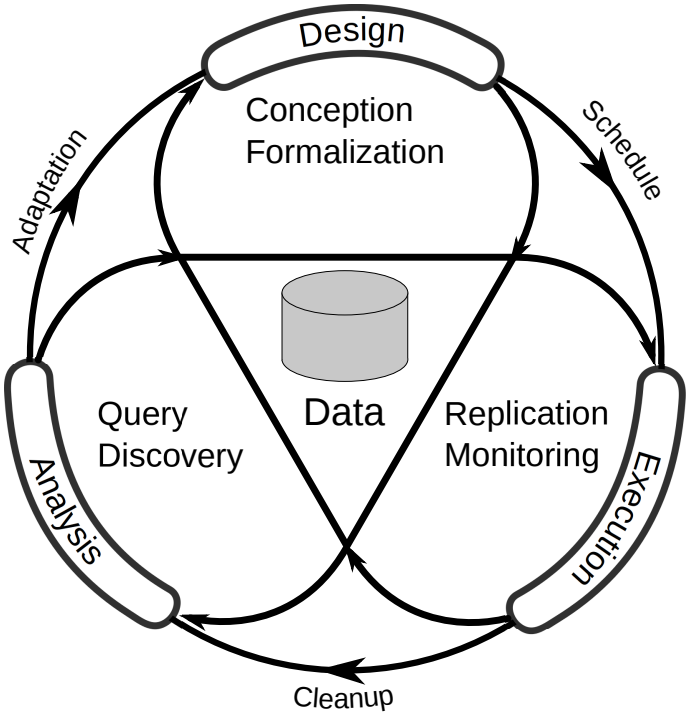


Figure 3.2.1.: Scientific workflow for a network experiment. The life cycle of an experiment consists of three phases: design, execution, and analysis.

others. This can be achieved by providing tools that assist the researcher throughout the experiment workflow as described next.

3.2.2 Experiment workflow

In general, every experiment consists of the following three phases:

- Design
- Execution
- Analysis

Experimentation is usually an iterative process resulting in a feedback loop between the analysis and design phases as depicted in Figure 3.2.1. Results of a first version of an experiment might reveal critical parameters in the design of the experiment. Based on these results, the experiment can be fine-tuned.

Experiment design

Every experiment starts with an idea. The idea is then developed into an fine-grained experiment description with clear instructions for every single step. For this process, a domain expert with profound knowledge of the research subject is required. The *parameters* for the particular experiment have to be specified. For networking experiments, these

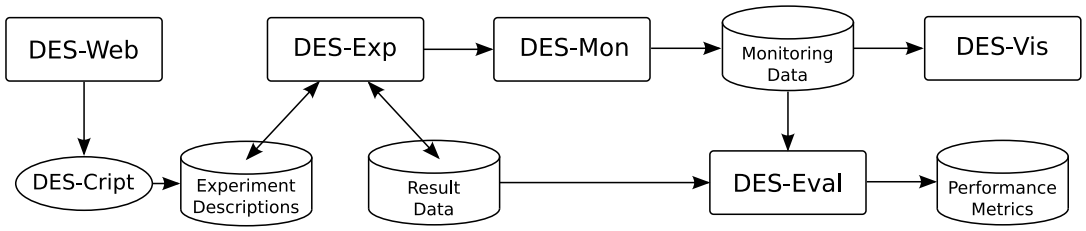


Figure 3.2.2.: Architecture of the DES-Testbed Management System (DES-TBMS).

parameters usually include the number of network nodes, their location, traffic patterns, and many more. The resulting parameter space may quickly become very large. Methods for the *Design of Experiment* (DoE) should be used to minimize the parameter space and thus reduce experiment run time [93].

Experiment execution

Once an experiment has been designed and formally described, it is ready to be scheduled for execution. The experimentation environment has to be in a *defined state* before and after the execution of the experiment. For this reason, a *configuration* phase precedes the experiment execution and a *cleanup* phase follows after the execution. Usually, an experiment is replicated several times in order to obtain statistically sound results. For this reason, a tool that allows the automated execution of experiments proves helpful for the researcher. *Domain-specific languages* (DSL) have been used to develop a formalized and machine-readable experiment description languages for automated experiment executions [94, 95]. Throughout the experiment, the state of the testbed nodes is monitored in order to detect unexpected behavior such as crashes and reboots.

Experiment analysis

Finally, the raw data produced by the experiment is processed to extract the prior defined performance metrics. The performance metrics are then investigated to validate or invalidate the hypothesis of the experiment. Based on the results, the original experiment may be fine-tuned or adapted and eventually rescheduled for execution.

3.2.3 Testbed Management System (TBMS)

The *DES-Testbed Management System* (DES-TBMS) [96] comprises tools to support the design, automated execution, and analysis of experiments on the DES-Testbed. The software framework consists of six components, each dedicated to a specific task in the experiment workflow. The architecture and the relationships between the components are depicted in Figure 3.2.2, a description of each component follows.

DES-Cript is a *domain specific language* (DSL) based on XML, which defines and describes network experiments in a holistic way [95, 97]. As the structure of the underlying network is abstracted, DES-Cript is not limited to the DES-Testbed. Each DES-Cript file

contains a general information section, followed by the available network nodes assigned into groups with particular roles. Next, actions are assigned to a group or individual nodes. As existing DES-Cript experiment descriptions can be edited and reused, they can be used to isolate critical parameters. Moreover, DES-Cript files provide a well-defined experiment documentation without any further effort. DES-Exp provides an experiment manager which is responsible for the scheduling and execution of experiments defined in the DES-Cript experiment description. DES-Exp also assures a defined state at the beginning of each experiment and its replications. DES-Web provides a web interface to DES-Exp, which allows to create, modify, and schedule experiments using DES-Cript. The network monitoring tool DES-Mon is based on SNMP and retrieves the network state from the DES-Nodes. DES-Mon collects data from the wireless interfaces, the kernel routing table and data from the sensor nodes.

DES-Vis is a 3D-visualization software based on the JavaView framework [98] to display gathered data obtained from experiments or to show the current state of the network. For routing algorithms it can display the existing links between the network nodes or color the links and nodes for the evaluation of channel assignment algorithms. The evaluation tool DES-Eval enables the post-processing of the experiment results supporting workflows for an automatic evaluation process. It provides several configurable input, processing, and output modules. In addition, new modules can easily be written and integrated. Existing modules support input from log files or database records, statistical analysis using R [99], and output as plotted graphs.

3.3 Characteristics of the network topology and channel conditions

This sections covers several experiments that have been carried out on the DES-Testbed in order to validate common assumptions of WMNs. The experimentally determined channel characteristics are an important input for channel assignment algorithms. For instance, *adjacent channel interference* (ACI) can be measured and thus the existence of possible non-interfering channels can be validated.

3.3.1 Network topology and link quality

As a first experiment, we determine the network topology of the DES-Testbed with the number of existing links and their corresponding quality. As link metric, we use the *expected transmission count* (ETX) [100]. The measurements were performed with the ETX daemon of the DES-Chan framework as described in Section 6.2. We also investigate the impact of the particular channel on the network topology. For this, we determine the number of links and their respective ETX values separately on each channel of the 2.4 GHz and 5 GHz frequency band. It is expected that the signal range decreases with a higher frequency [36]. Therefore, the quality of links on channels at 2.4 GHz is likely to decrease when the channel is switched to 5 GHz. The experimental validation is important since a link that exists

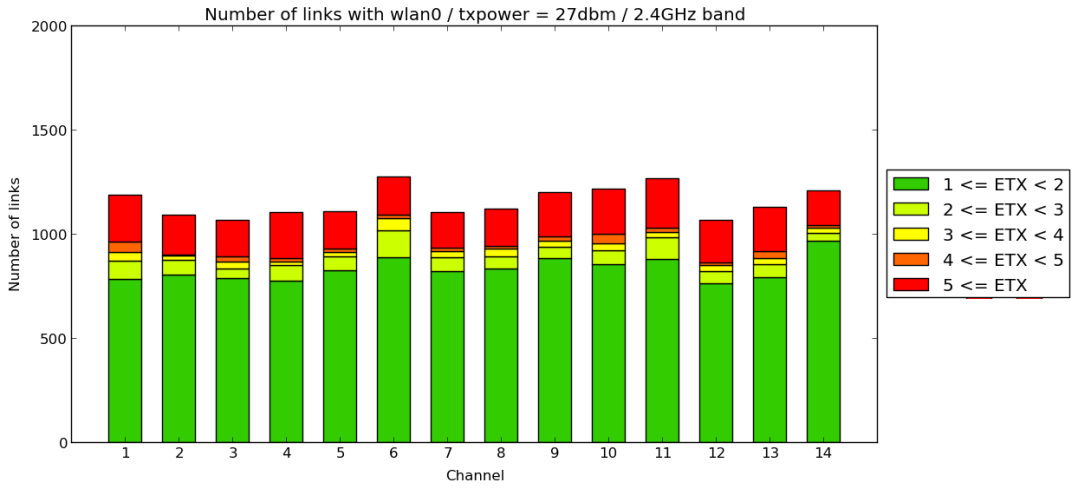


Figure 3.3.1.: Number of links and their quality on the 2.4 GHz frequency band using the Ralink USB WNIC with a transmission power of 27 dBm. About 1100 links have been detected per channel, the majority of them are of high quality with $\mathsf{ETX} < 2$.

when using a channel on the 2.4 GHz band may not exist when the corresponding WNICs are tuned to a channel of the 5 GHz spectrum. If a channel assignment algorithm does not consider this constraint, the network connectivity may be affected after the channel assignment procedure.

We measured the number and quality of links using all available 128 DES-Nodes. For the experiments, we tuned one WNIC on each mesh router to the same channel. We started the ETX daemon to determine the amount of links and the corresponding link quality for each network node. The ETX values were measured periodically for 3 minutes on each channel with a sliding window of 10 seconds and an beacon interval of 1 second. We executed the measurements for both WNIC types of the DES-Nodes, the IEEE 802.11b/g Ralink USB adapter which is attached to a 2.4 GHz antenna and the IEEE 802.11a/b/g Atheros-based MiniPCI card which is attached to a dual-band antenna.

Figure 3.3.1 shows the results using the Ralink WNIC on 2.4 GHz with a transmission power of 27 dBm as reported by the driver. We can observe that on all channels we have about 1100 unidirectional links. The majority of the links are of high quality, expressed by $\mathsf{ETX} < 2$. The number of medium quality links for which $2 \leq \mathsf{ETX} < 5$ is very low. Low quality links expressed by $\mathsf{ETX} > 5$ have been observed more often.

Figure 3.3.2 shows the results of the same experiment using the Atheros WNIC, also with the transmission power set to 27 dBm. On each channel, we observe roughly the same number of links, however, the absolute number of links compared to the previous experiment is much higher, in average it is about 1800. The results are similar in that the majority of links is of high quality. Again, medium quality links for which $2 \leq \mathsf{ETX} < 5$ are rare. The absolute number of links is about 60 % higher compared to the results with the Ralink network adapter. The results of a measurement using the Atheros card with the transmission power set to 17 dBm is shown in Figure 3.3.3. With the lower signal power,

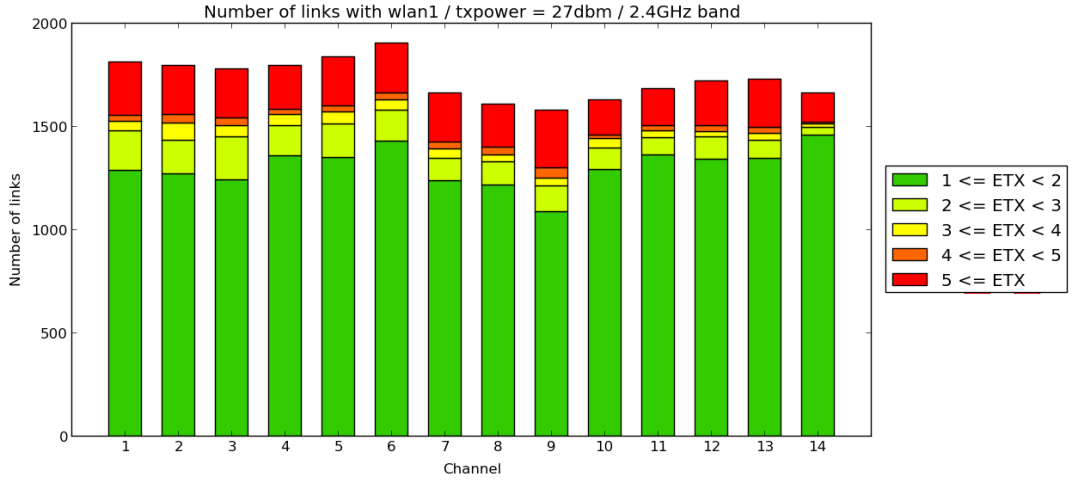


Figure 3.3.2.: Links and their quality on the 2.4 GHz frequency band using the Atheros MiniPCI WNIC with a transmission power of 27 dBm. The number of links is significantly higher compared to the previous experiment using the Ralink adapter.

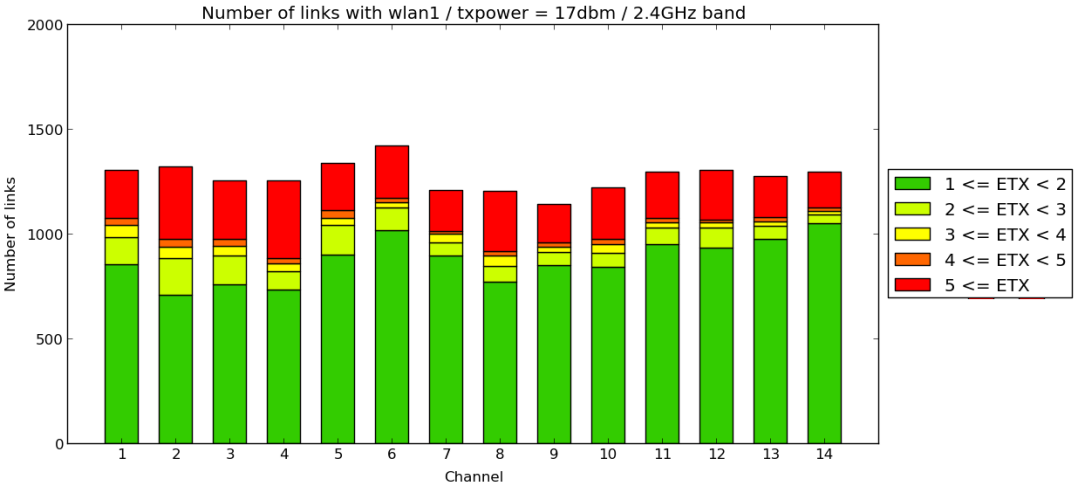


Figure 3.3.3.: Links and their quality on the 2.4 GHz frequency band using the Atheros MiniPCI WNIC with a transmission power of 17 dBm. With the reduced transmission power, the results are similar to the ones in the first experiment using the Ralink adapter.

we get about the same number of links as in the first experiment with the Ralink adapter.

Figure 3.3.4 shows the results using the Atheros WNIC on 5 GHz with a transmission power of 27 dBm. Again, the majority of the links are of high quality, whereas the number of medium quality links is very small. The amount of links and their quality does not vary a lot among the particular channels of the 5 GHz spectrum, except for channel 44. The carrier sensing statistics revealed a high channel congestion on this particular channel caused by external devices that leads to the lower number of links. The expected decrease of the number of links for the higher frequencies can be observed, in average we count about 750 links per channel.

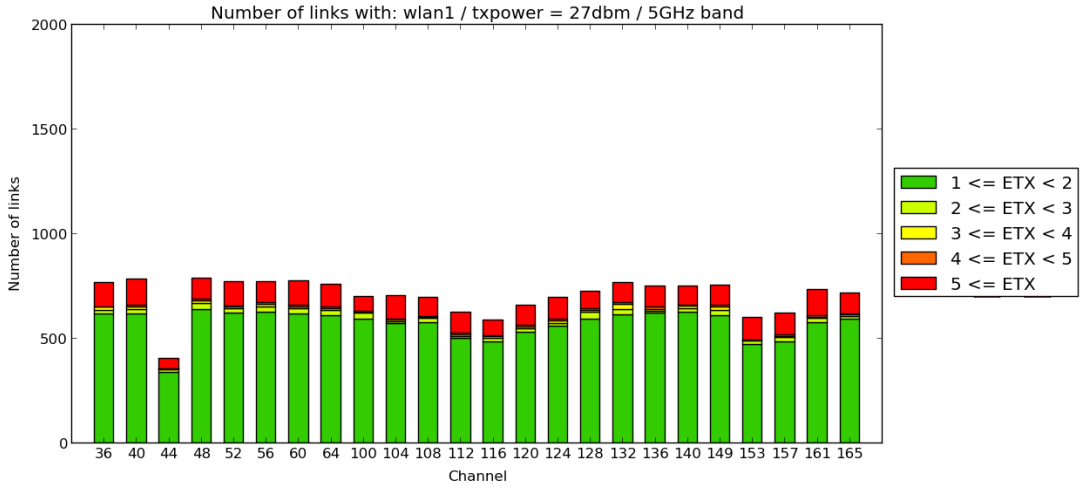


Figure 3.3.4.: Number of links and their quality on the 5 GHz frequency band. We observe significantly fewer links compared to the experiments on the 2.4 GHz band.

In order to investigate the effects of the channel usage to the network topology, we take a closer look at the high quality links with ETX < 2 throughout all four experiments. We observed, that the number of links in the whole network depends on the utilized WNIC, the frequency band, and on the transmission power. The highest number of links have been observed using the Atheros-based WNIC with a transmission power of 27 dBm on the 2.4 GHz band. Setting the transmission power to 17 dBm results in a similar number of links compared to the experiment with the Ralink adapter. The fewest number of links have been reported with the Atheros interface on the 5 GHz band, which is as expected. The experiments validate the assumption that the network topology is dependent on the particular frequency band in use and also on the particular wireless network adapter. Therefore, it can not be assumed that the link quality or even the network connectivity stays the same if channels are switched without considering the transmission power and the frequency band.

3.3.2 Channel conditions

As next, we investigated the conditions regarding the throughput of all available channels in the 2.4 and 5 GHz band. We selected 40 high quality links with ETX $= 1$ for each channel and measured the throughput on these links sequentially. This way, no parallel measurements were carried out in order to avoid interference effects. For the throughput measurements, we used `iperf` and generated UDP traffic for 30 seconds with a data rate of 6 MBit/s which corresponds to the data rate settings of the WNICs.

The results for the throughput measurements in the 2.4 GHz band are shown in Figure 3.3.5. The throughput of channel 14 is close to the theoretic maximum with low deviation. Channel 14 is exclusively used in the DES-Testbed due to regulatory constraints which we do not follow in our experimental testbed (see Section 3.1). The lower results for the remaining channels can be credited to the co-located WLAN networks on our campus,

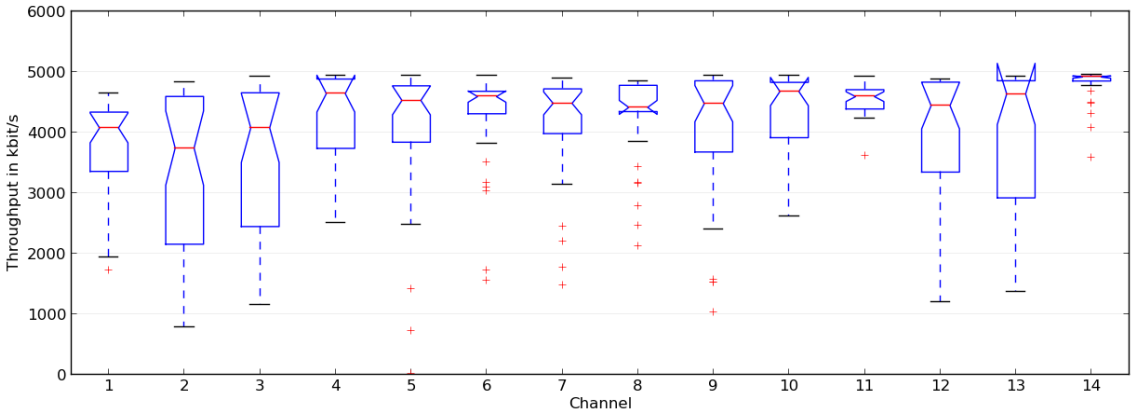


Figure 3.3.5.: Througput per channel on the 2.4 GHz band.

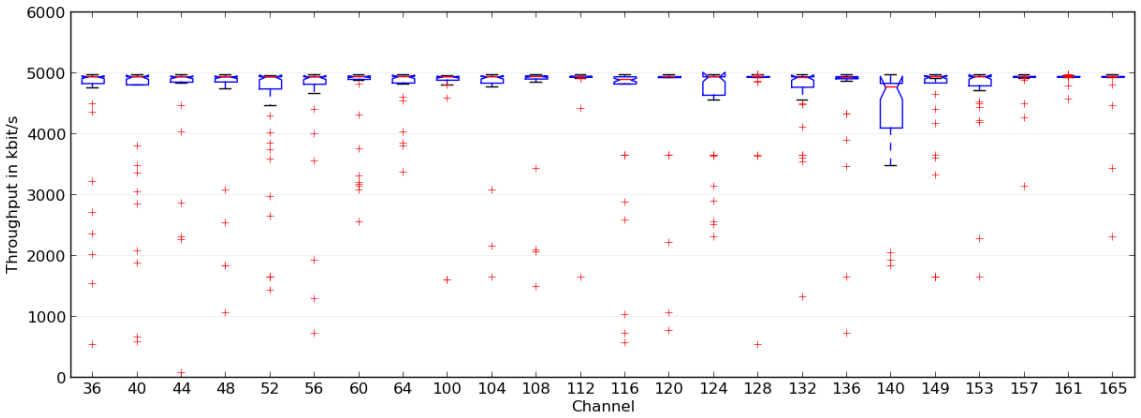


Figure 3.3.6.: Throughput per channel on the 5 GHz band.

which operate on the channels 1, 6, and 11. Also, the effects of adjacent channel interference are visible for the neighboring channels. As a result, all channels but channel 14 suffer from external interference in the 2.4 GHz band.

The results for the channel conditions in the 5 GHz band are shown in Figure 3.3.6. Compared to the results of the 2.4 GHz channels, the achievable throughput of all channels is close to the expected maximum. This can be credited to the fact, that the co-located WLAN networks do not operate on those channels. As a result, the levels of external interference is much lower on the channels in the 5 GHz band than in the 2.4 GHz band and thus should be preferred for experiments.

3.3.3 Adjacent Channel Interference (ACI)

IEEE 802.11b/g offers three non-overlapping channels, for instance {1, 6, 11}, and all available channels in IEEE 802.11a are non-overlapping. This means in theory, that concurrent transmissions on these channels should not interfere with each other. In practice, experiments on different experimental platforms have shown, that the non-interfering characteristics do not hold for many reasons [36,54,72,101–103]. The causes for this effect are board crosstalk, radiation leakage, and a small distance between antennas of simultane-

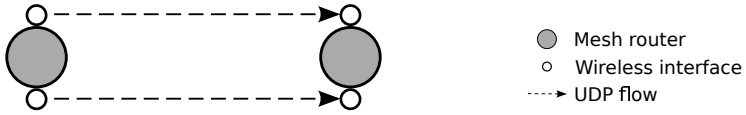


Figure 3.3.7.: Experiment setup for measuring the effect of adjacent channel interference on adjacent traffic flows.

ously active radios. Among them, antenna distance has the most severe effect on the performance [104]. To avoid the near-antenna effect, the experimentally specified minimum distance between two antennas is about 1 m. Since mesh routers are usually more compact, it is almost impossible to design a multi-radio mesh router with sufficient antenna distance. This is also the case with DES-Nodes, on which the three WLAN antennas are mounted with a distance of about 30 cm. Therefore, we also expect side-effects on the theoretical non-interfering channels, which are subject to experimentation in this section.

One of the proposed measurement-based interference estimation schemes is the *link interference ratio* (LIR) [46]. For two links $l_{u,v}$ and $l_{x,y}$ the LIR is defined as

$$\mathrm{LIR}_{l_{u,v},l_{x,y}} = \frac{T_{l_{u,v}}^{l_{u,v},l_{x,y}} + T_{l_{x,y}}^{l_{u,v},l_{x,y}}}{T_{l_{u,v}} + T_{l_{x,y}}}$$

where $T_{l_{u,v}}$ is the unicast throughput for link $l_{u,v}$ when only this link is active and $T_{l_{u,v}}^{l_{u,v},l_{x,y}}$ is the unicast throughput for the link when the link $l_{x,y}$ is active simultaneously. The LIR expresses the impact of interference two links exert on each other by relating the aggregate throughput of both links when they are active individually to the aggregate throughput when they are active simultaneously. A LIR value of 1 indicates that the two links do not interfere at all, whereas a LIR value of 0.5 means that the aggregate throughput is halved when both links are active at the same time.

The LIR is suitable to investigate the impact of the channel distance on two simultaneous transmissions. Therefore, in a first experiment we measure the LIR of two links being adjacent to the same node for a varying channel distance to investigate the effect of *intra-path* interference. In a second experiment, two links with different sender and receiver pairs which are in each others interference range were chosen. This experiment will give insights on the impact of *inter-path* interference. For both experiments we use two Atheros MiniPCI cards with the `ath5k` driver. The auto data-rate algorithm is used and RTS/CTS disabled. The channels of the links are sequentially set to all possible combinations on each frequency band. We perform the experiment for all channel combinations of the 2.4 GHz and frequency band, the 5 GHz frequency band, and finally using both bands simultaneously. The experiment has been repeated 40 times for each channel distance.

Adjacent traffic flows

Based on the network topology of the DES-Testbed, we select a subset of 5 high quality links with an ETX value of 1. For each node pair, we selected one node as sender and the other as receiver, as depicted in Figure 3.3.7. In order to measure the LIR we generate two

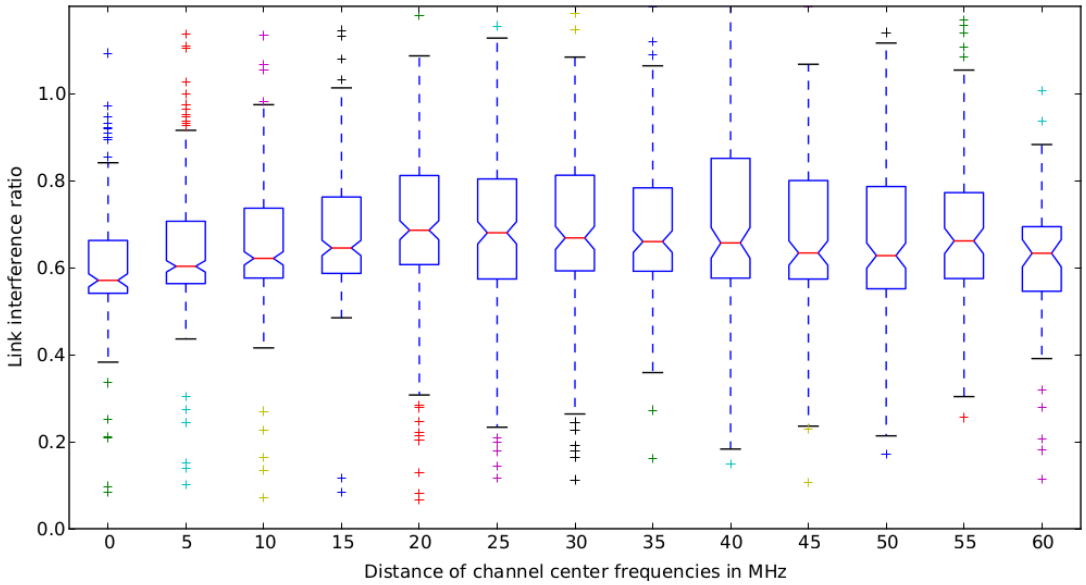


Figure 3.3.8.: Results of the LIR of adjacent flows for channel combinations on the 2.4 GHz band. The LIR of two links in respect to their spectral distance is shown. The median for all channel combinations is about 0.6.

UDP unicast flows from one of the routers (sender) to the other (receiver). Each flow is generated with `iperf` with a data rate of 54 MBit/s for 30 seconds. After the sequential flows, we start both flows another time simultaneously. We measure the individual and aggregate throughput and compute the LIR. We chose this scenario because it is common in multi-hop WMN where a multi-radio node utilize more than one radio at the time in order to increase the overall throughput.

Results for the 2.4 GHz band

The results for the channel combinations on the 2.4 GHz band are depicted in Figure 3.3.8. Unfortunately, they show that in the DES-Testbed no pair of channels of the 2.4 GHz band are non-interfering for adjacent traffic flows. The median of the LIR values is about 0.6 regardless of the used channel combination, which means that the aggregate throughput is almost halved when two interfaces are activated simultaneously. Concluding from the results, a channel assignment with the highest possible spectral distance would only lead to a minor increase of the throughput. As already mentioned, we credit these results to the near-antenna effect of the DES-Nodes.

Results for the 5 GHz band

The results for a subset of all possible spectral channel distances of the 5 GHz band are depicted in Figure 3.3.9. It can be observed that the median of the LIR value increases with channel distances of up to 180 MHz. This rise of the LIR is much slower than expected, but the results show that the median is about 0.8 for a channel distance of at least 80 MHz. For a channel distance of 320 MHz and more the LIR decreases again, which we did not

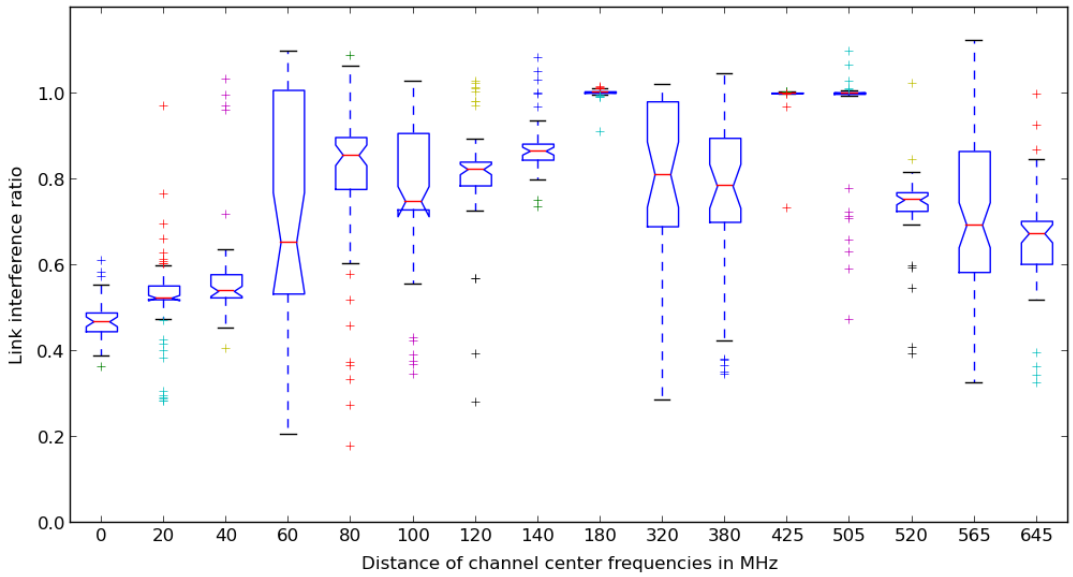


Figure 3.3.9.: Results of the LIR of adjacent flows for channel combinations on the 5 GHz band. The median of the LIR value increases with channel distances of up to 180 MHz and then decreases again.

expect. In a first investigation, this seems to be related to the link quality. We used `tcpdump` to monitor the *received signal strength indication* (RSSI) values for each correctly received frame. We observed that the lower the RSSI values are, the lower the LIR values are for a increasing channel distance.

Results for both frequency bands

In the last set of experiments for adjacent flows, we selected only channel combinations from both available 2.4 GHz and 5 GHz frequency bands. One sender/receiver pair of WNICs is tuned to channel $c_1 \in \{1, 13\}$ whereas the other is tuned to channel $c_2 \in \{36, 64, 100, 140, 149, 165\}$. The results, as depicted in Figure 3.3.10, show that the median of the LIR is between 0.8 and 1 and therefore only a small decrease of performance can be observed.

Discussion of the results

Unfortunately, the results of the experiments differ vastly from the theoretical assumptions. For all channel combinations using only the 2.4 GHz band, a LIR of about 0.6 was measured, which is only a minor improvement to the single channel network scenario. Minor interference effects are observed with a channel distance of at least 80 MHz on the 5 GHz band. Therefore, two simultaneously active flows should make use of both frequency bands, where a LIR of about 0.8 was measured.

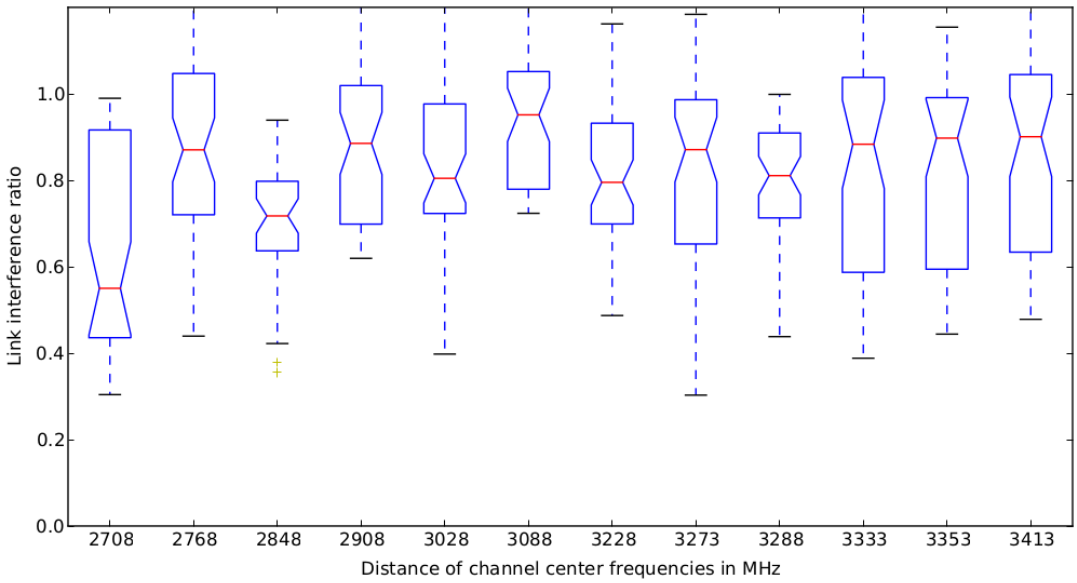


Figure 3.3.10.: Results of the LIR of adjacent flows for channel combinations on the 2.4 GHz and 5 GHz bands. The median of the LIR for all channel combinations is about 0.8.

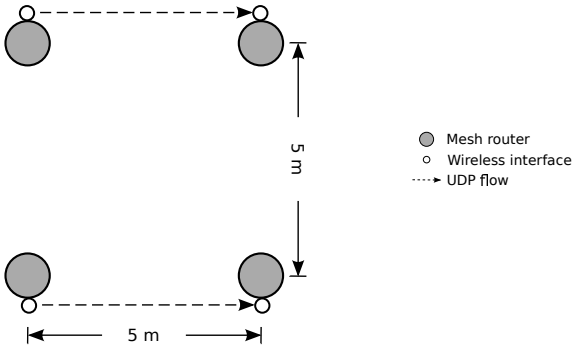


Figure 3.3.11.: Experiment setup for measuring the effect of adjacent channel interference on the LIR with non-adjacent traffic flows. Two node pairs are selected which are located in the same room.

Non-adjacent traffic flows

In this study, we measure the LIR for two non-adjacent flows. For this, two pairs of DES-Nodes located in a single room are used. The experiment setup is depicted in Figure 3.3.11. The experiment steps and parameters are the same as in the previous experiment.

Results for the 2.4 GHz band

The results for the channel combinations on the 2.4 GHz band are depicted in Figure 3.3.12. With a channel distance of 30 MHz and more, the median of the LIR is usually above 0.8 which implies that a significant higher throughput can be achieved with a channel distance of at least 30 MHz.

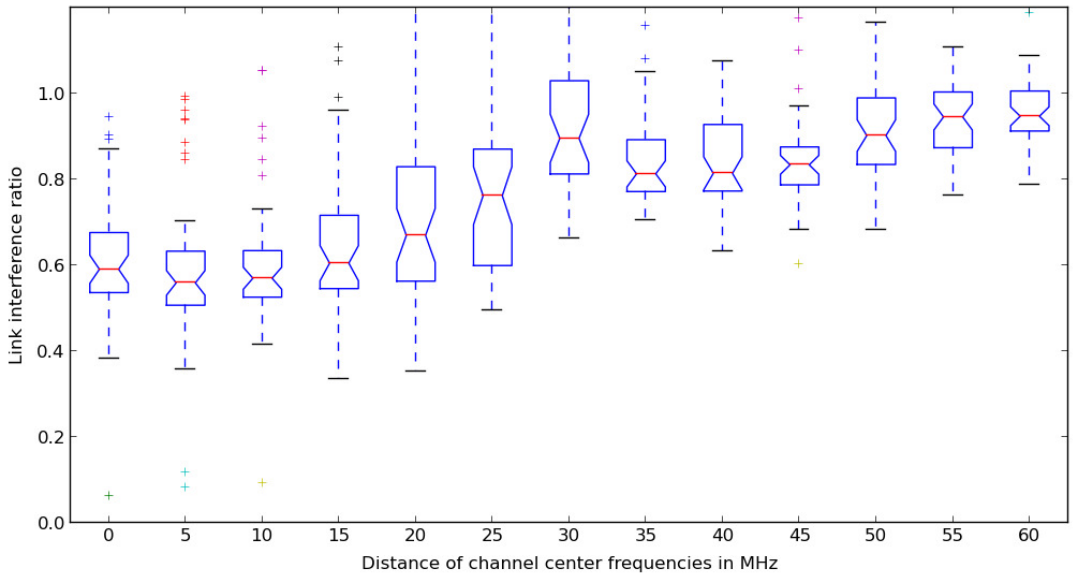


Figure 3.3.12.: Results of the LIR of non-adjacent flows for channel combinations on the 2.4 GHz band. With a channel distance of 30 MHz, the median of the LIR is usually above 0.8

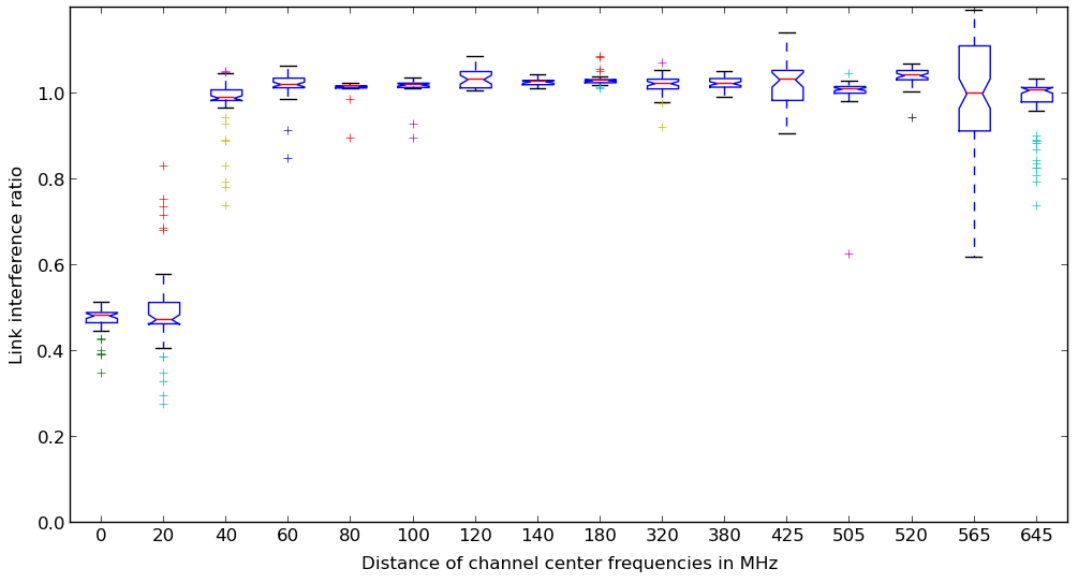


Figure 3.3.13.: Results of the LIR of non-adjacent flows for channel combinations on the 5 GHz band. From a channel distance of 60 MHz, the median of the LIR is close to 1.

Results for the 5 GHz band

The results for the 5 GHz band are depicted in Figure 3.3.13. From a channel distance of 40 MHz, the median of the LIR is close to 1, which means that there are hardly any interference effects.

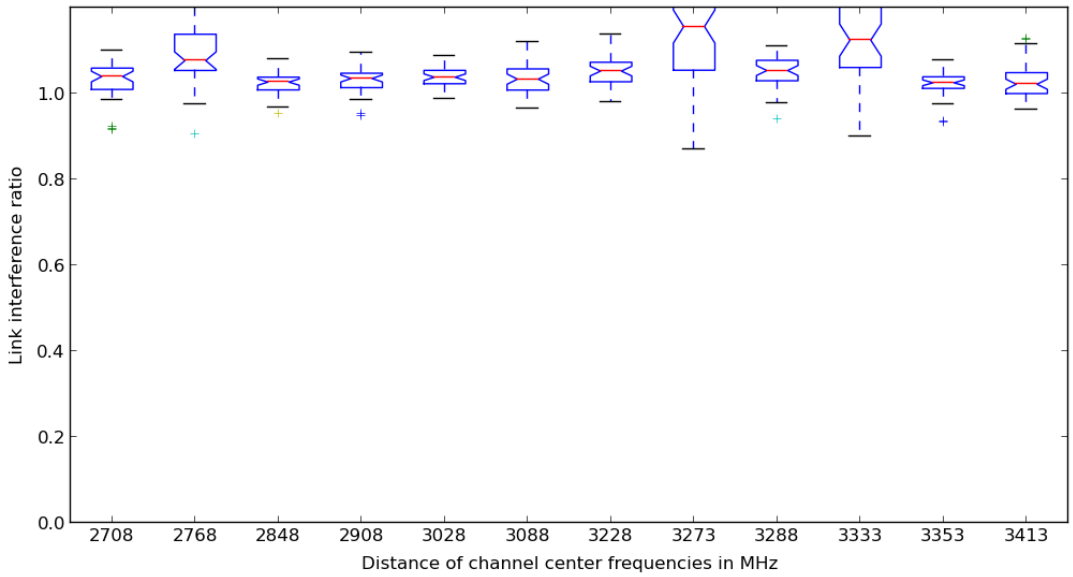


Figure 3.3.14.: Results of the LIR of non-adjacent flows for channel combinations using the 2.4 GHz and 5 GHz bands. The median of the LIR is close to 1.

Results for both frequency bands

For the last experiment, we selected only channel combinations from both available 2.4 GHz and 5 GHz frequency bands. The results, as depicted in Figure 3.3.14, show that the median of the LIR is close to 1 for all channel combinations and therefore no interference effects are observed.

Discussion of the results

Although none of the channel combinations in the 2.4 GHz band allow completely non-interfering transmissions, a minimum channel distance of 30 MHz should be used for simultaneously active flows in order to achieve the highest possible throughput. This results in three possible channels $\{1, 7, 13\}$ for an efficient channel assignment for non-adjacent flows. On the 5 GHz band a minimum spectral channel distance of 40 MHz is sufficient to experience only negligible interference effects. Using both bands simultaneously, hardly any interference effects could be measured with the median of the LIR being around 1.

The results for non-adjacent flows show fewer impact of interference as the corresponding experiments with adjacent flows. We assume the main causes for these results being the greater antenna distance for the experiments with non-adjacent flows (5 m to 0.3 m).

3.3.4 Multi-hop path interference

In this experiment, we validate the gained knowledge about the channel characteristics on the DES-Testbed with a manual channel assignment. For this, we create a chain topology of 5 mesh routers, on which we start traffic flows over up to four hops. First, we apply the same channel to all links in the chain topology, thus creating a single channel network

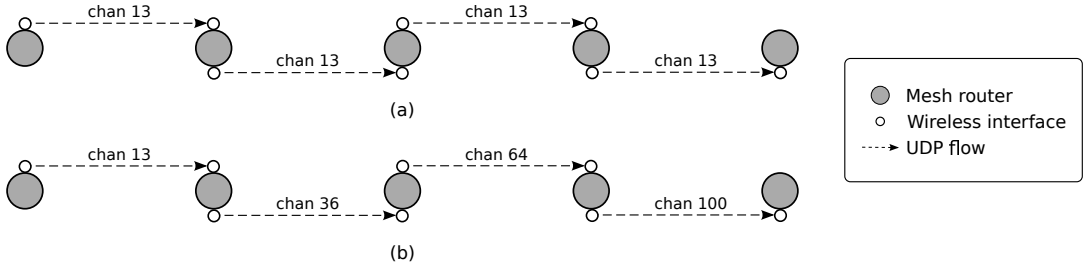
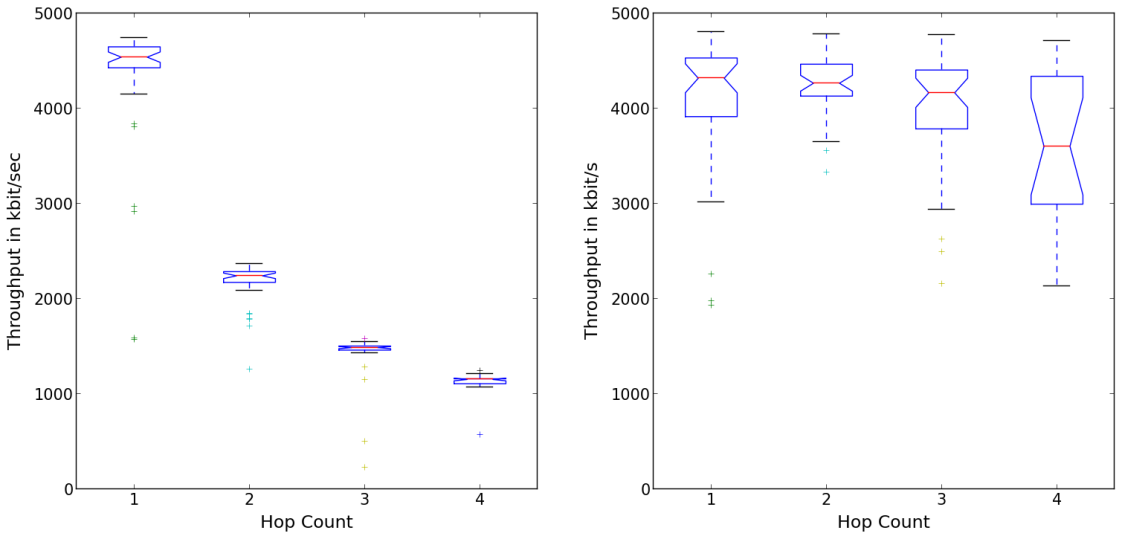


Figure 3.3.15.: Experiment setup to measure throughput in a single- and multi-channel network. A subset of the mesh routers is selected to create a chain topology. In (a), we apply the same channel to all links, thus creating a single channel network scenario. In (b), we manually assign channels which promise an increase of throughput based on the results of the previous experiments.

scenario as depicted in Figure 3.3.15 (a). We then start an UDP flow with `iperf` from the first node of the chain to the second node. Afterwards we start an UDP flow from the first node to the third and so on, up to the last node in the chain. We configured the WNICs with the fixed data rate of 6 Mbit/s and send the UDP flow for 30 s with the same data rate. We repeat the experiment for each hop 40 times.

The results for the single channel network scenario for the throughput in relation to the hop count are depicted in Figure 3.3.16a. As expected for the single channel network case, the throughput is more then halved on the first hop and keeps dropping with an increasing hop-count. Based on the results of the channel characteristics experiments, we then manually assign channels in a way, that promises the biggest decrease of interference effects. As observed from the experiments on adjacent flows, both frequency bands should be used for the respective WNICs. We expect the throughput to be significantly higher compared to the single channel network. We apply the channels $\{13, 36, 64, 100\}$ to the links as depicted in Figure 3.3.15 (b) with which we expect to exhibit only minor interference.

The results for the multi-channel network scenario are depicted in Figure 3.3.16b. As expected, the manual channel assignment leads to a higher throughput if the hop-count is bigger than 1. It only drops slightly with the increasing hop-count which implicates that the interference effects have been reduced significantly with the chosen channel assignment. These results show that the experimentally determined channel characteristics also hold in multi-hop scenarios and underline the potential performance gain that can be achieved by an efficient channel assignment.



(a) Results for the single-channel path experiment. (b) Results for the multi-channel path experiment.

Figure 3.3.16.: Results of the multi-hop path interference experiment. For the single channel network case, the throughput is more than halved on the first hop and keeps on dropping with an increasing hop-count as depicted in (a). With the manual channel assignment, the throughput can be significantly increased as shown in (b).

3.4 Discussion

The experiments were performed to gain insights on the network topology and the channel characteristics in the DES-Testbed. We first showed, that the network topology and the quality of network links depend on the utilized WNIC and the frequency band. In a second series of experiments, the effects of ACI have been investigated. The LIR of two links is significantly lower for adjacent than for non-adjacent traffic flows. However, using channels on both frequency bands also promises a higher throughput for adjacent traffic flows.

The results of the experiments have been validated with a manual channel assignment in a chain topology spanning 4 hops. The throughput has been significantly increased compared to the single channel network scenario. A comparison of the experiment results to the common assumptions of channel assignment algorithms yields some interesting deviations. First, the assumption of orthogonal channels as theoretically offered by IEEE 802.11 and considered in many channel assignment algorithms does not hold in practice. In contrast, if the experimental results indicate that the actual number of available channels is significantly reduced which may affect the performance of the algorithms. Second, channel assignment algorithms usually assume that a link between two nodes or radios exist or not throughout the network operation. However, the experiments have shown, that the link quality may depend also on the WNIC characteristic and on the used frequency band. Therefore, when designing channel assignment algorithms, it should be considered that channel switches may alter the link quality or even the network connectivity.

CHAPTER 4

Methodology

This chapter covers the methodology of the experimental studies performed for this dissertation. The applicability of common performance evaluation methods in computer science to the domain of channel assignment is discussed and examples are given how the performance of channel assignment algorithms has been measured in previous works. Subsequently, the *channel assignment benchmark* (CAB) is presented, comprising domain-specific performance metrics and scenarios for the performance evaluation of channel assignment algorithms in wireless testbeds. Excerpts of this chapter have been published in [105].

4.1 Motivation

The methodology of this dissertation is located in the domain of empirical and experimental research in large-scale wireless networks. Developing implementation candidates of channel assignment algorithms and evaluating their performance in real network environments, has thus gained high significance. Several design goals have to be taken into account for an holistic experiment methodology, the most important ones are the following:

- *Performance comparison of algorithms:* The methodology should allow to compare the performance of a wide range of algorithms, thus giving insights on the feasibility of different strategies.
- *Portability across different experimentation environments:* The methodology should be generic to a degree that it can be easily ported. This way, the performance of channel assignment algorithms can be compared across multiple testbeds with different types of network nodes and network topologies.
- *Minimize experiment run time:* Since experiments in wireless testbeds are executed in real-time and are labor-intensive, a solid and sound experimentation methodology is required to minimize the testbed usage while still retrieving meaningful results. Efficient performance metrics and scenarios are required to accomplish this task.

After briefly discussing common tools for the performance analysis, we introduce the channel assignment benchmark (CAB). CAB has been designed to meet the introduced design goals thus allowing a comparison of different channel assignment approaches across different experimentation facilities.

4.2 Performance evaluation methods

4.2.1 Benchmarking

Benchmarking is a well known method in computer science for the performance evaluation of a *system under test* (SUT) [106,107]. For the evaluation, usually a program or a suite of programs is run on the SUT under reference conditions. Performance metrics are defined together with scenarios that allow to measure these metrics. The performance score of a SUT can be compared intuitively with the numbers scored by other systems, which allows a comparison of the performance of different solutions.

4.2.2 Performance evaluation of channel assignment algorithms

Many different methods have been used for the performance evaluation of channel assignment algorithms in previous works. This comprises using different measurement procedures and performance metrics, different experimentation environments, and different tools and programs (i.e., network traffic generators). For these reasons, the comparison of the performance based only on the published results alone is limited.

In general, the goal of the performance evaluation in this domain is to measure the reduction of interference effects due to the channel assignment. The developed performance metrics that have been used in previous works can be distinguished into two classes. The first class comprises performance metrics that measure the *decrease of interference* achieved by the channel assignment algorithms. One example for such a metric is the *fractional network interference* (FNI) metric, which has been defined in [36] and used for further approaches in [108]. The FNI is the ratio of the number of edges of the conflict graph [1,52] of the single channel network and the number of edges in the conflict graph after the channel assignment. The FNI is feasible for algorithms that model the interference with a conflict graph and gives an insight of the performance in terms of interference reduction according to the used interference model. However, the results are only comparable to algorithms that also model the interference with a conflict graph.

The second class of performance metrics avoids this limitation by measuring the *gain in network performance* after the channel assignment. One common metric of this class is the network saturation (or network capacity) which describes the maximum load the network can carry and which is also used in other domains for performance evaluation. For channel assignment algorithms, the network saturation can be measured and compared for different algorithms or the single channel network as a baseline.

Another challenge for the comparability of the performance is that the algorithms are evaluated in different experimentation environments, comprising analytical analysis, simulation studies, and experiments in real network deployments such as wireless testbeds. Recently, experimentation in wireless testbeds has gained in importance, since often-used simplified interference models, such as the *m-hop interference model* [35], have been proven to be not very accurate for real network deployments [46, 49, 109]. However, performance measurements on different testbeds can usually not be compared directly. Characteristics of the particular testbeds have to be taken into account, such as the network topology and the network environment. Additionally, the testbeds may differ in number of nodes, number of network interfaces per node, and the number of available orthogonal channels.

4.3 Channel Assignment Benchmark (CAB)

The *channel assignment benchmark* (CAB) has been developed for the performance evaluation of channel assignment algorithms in wireless testbeds. CAB comprises performance metrics in respect to the domain-specific goal to reduce interference and scenarios to measure the performance of channel assignment algorithms based on these metrics. In the channel assignment domain, the SUT comprises the particular channel assignment algorithm and the wireless testbed on which the performance evaluation is carried out on. The developed performance metrics and the benchmark are not limited to a particular testbed and can be applied easily to other existing systems.

4.3.1 Performance metrics and scenarios

Considering channel assignment algorithms, metrics are of particular interest that express the performance in regard to the reduction of *inter-path*, *intra-path*, and *external* interference effects as described in Section 2.2. In this section, we present performance metrics that assess the reduction of interference by measuring the gain in network performance with sequential and simultaneous throughput measurements of possible interfering links. Additionally, the network saturation is a feasible indicator of the increased network capacity. Also of interest is the introduced protocol overhead in form of exchanged control messages that is imposed on the network by the channel assignment algorithm. We describe each performance metric in detail and a concrete scenario to measure the metric on the DES-Testbed.

Interference-free environment

Several of the developed performance metrics are based on measurements in an *interference-free* environment to determine the maximum throughput that is achievable on a link or path as a baseline. However, the DES-Testbed is distributed over several buildings with co-located networks operating in the same frequency spectrum. Therefore, the environment can not be guaranteed to be interference-free. The experiments for the channel conditions

showed that there is no radio activity on most of the channels of the 5 GHz in the environment of the DES-Testbed (see Section 3.3.2). However, a significant channel congestion has been determined for the channels of the 2.4 GHz spectrum. Therefore, we use the term interference-free in a way, that it comprises only the devices under our control, meaning that there is no additional radio activity from other testbed nodes during the measurement.

4.3.2 Intra-path Interference Ratio (IAR)

The throughput of a multi-hop data flow may be reduced due to the impact of intra-path interference. The goal of the *Intra-path Interference Ratio* (IAR) metric is to assess, to what degree the intra-path interference effects have been reduced with channel assignment. The metric is calculated as follows.

Let $p_{u,v}$ be the m-hop path, with $m > 2$, from node u to node v. Let L be the set of all links on $p_{u,v}$. In order to measure the impact of the intra-path interference on $p_{u,v}$, we perform the following two measurement steps. First, we measure the achievable throughput t_l^{seq} on each link $l \in L$ sequentially, meaning only one link is active at a time. The aggregate throughput over all measurements expresses the maximum throughput that is available on this path in an interference-free environment and can be written as

$$T_{seq} = \sum_{l \in L} t_l^{seq} \tag{4.1}$$

As next, we measure the throughput of all links in $l \in L$ simultaneously. The aggregate throughput over all measurements expresses the maximum throughput that is available on this path when all links are activated at the same time and thus exert the highest level of interference on each other. It can be written as

$$T_{sim} = \sum_{l \in L} t_l^{sim} \tag{4.2}$$

The *intra-path interference ratio* (IAR) metric is then defined as the ratio of the two aggregated throughput measurements with

$$IAR = \frac{T_{sim}}{T_{seq}} \tag{4.3}$$

A value for IAR close to 1 means that the aggregate throughput is not reduced when the throughput is measured on all links on $p_{u,v}$ simultaneously. In this case, the algorithm was capable to reduce the intra-path interference with the calculated channel assignment. A lower value for IAR expresses a higher impact of intra-path interference on $p_{u,v}$.

Breaking the path down and measuring the capacity of each link on its own has the advantage to avoid bottleneck effects. Otherwise, if we start a data flow from node u to node v over $p_{u,v}$, the throughput on each link is limited by the throughput of the predecessor link. By considering each link on its own, we avoid the effect of such bottleneck links.

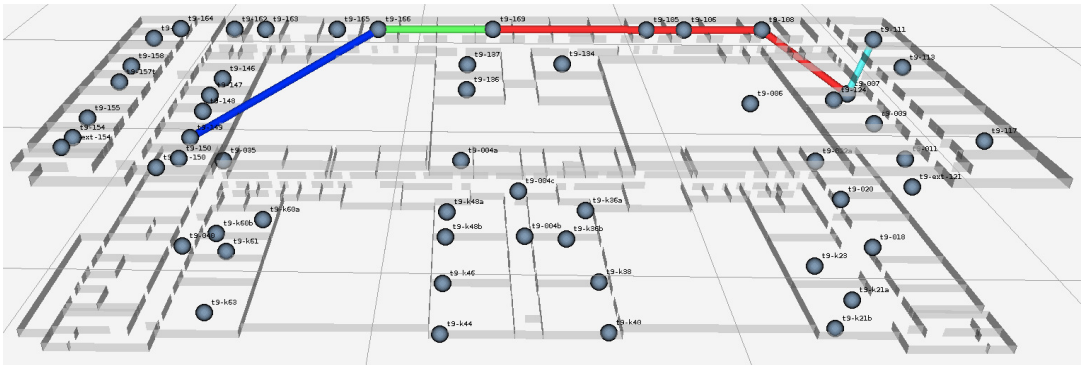


Figure 4.3.1.: Example of the link selection for the IAR measurements. The path $\mathfrak{p}$ consists of the following links: $\mathfrak{p}$ = ((t9-149, t9-166), (t9-166, t9-169), (t9-169, t9-108), (t9-108, t9-007), (t9-007, t9-111)).

Scenario

After the channel assignment procedure, a path of length 5 is selected using the *weighted cumulative expected transmission time* (WCETT) routing metric [72]. WCETT considers the spectral diversity on a path rather than just the shortest path based on the distance in the communication graph, thus choosing edges with a high spectral diversity. Choosing the links on the path with WCETT, allows us to exploit the spectral diversity created by the channel assignment algorithms as much as possible. We then start the measurement procedure with all links on the selected path as described above.

An example for such a selected 5-hop path is depicted in Figure 4.3.1. Only the links on that particular path are displayed, each color of a link represents a different channel.

4.3.3 Inter-Path Interference Ratio (IRR)

To measure the effect of inter-path interference, we select a subset L of all wireless links in the network with the additional constraint that every network node can only be adjacent to one link $l \in L$. This way, no multi-hop path is possible considering only the links in L. The links in L should be selected in a way that they are spatially close to each other, thus exerting a high degree of interference on each other when activated simultaneously. In a first step, we measure the achievable throughput t_l^{seq} on each link $l \in L$ sequentially, the aggregate throughput over all links is

$$T_{seq} = \sum_{l \in L} t_l^{seq} \tag{4.4}$$

In a second step, we activate all links simultaneously and measure the aggregate throughput which is

$$T_{sim} = \sum_{l \in L} t_l^{sim} \tag{4.5}$$

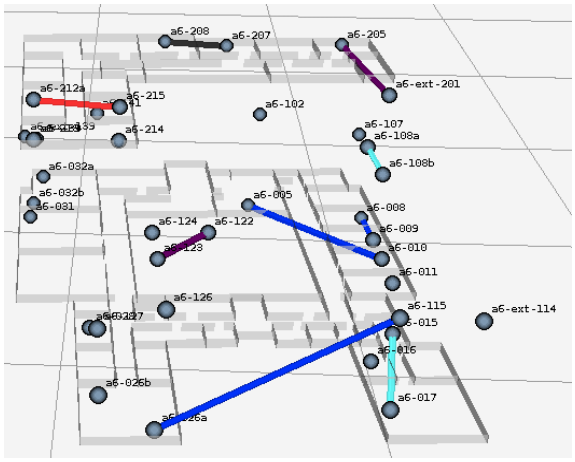


Figure 4.3.2.: Example of the link selection for the IRR measurements. The example shows 10 links selected in the building of Arnimallee 6.

The *inter-path interference ratio* (IRR) metric is then defined as the ratio between the aggregated throughput of sequential transmissions and concurrent transmissions.

$$\mathrm{IRR} = \frac{\mathrm{T_{sim}}}{\mathrm{T_{seq}}} \tag{4.6}$$

The simultaneous transmissions will exert a maximum interference, whereas the sequential transmissions express the achievable throughput in an environment free of inter-path interference. Since each network node is participating in at most one traffic flow, we measure the throughput reduction caused only by inter-path interference. A value for IRR close to 1 means that the aggregate throughput is not reduced when all links are activated simultaneously. Thus, the algorithm is capable to reduce the inter-path interference with the applied channel assignment. A lower value for IRR expresses that there is a higher impact of inter-path interference by multiple data flows.

Scenario

The DES-Testbed is distributed over four buildings on the computer science campus at Freie Universität Berlin. In order to ensure the spatial proximity when selecting a subset of 10 wireless links we constrain the choice of links on only one building. The mathematical institute located at Arnimallee 6 hosts 36 DES-Nodes. Considering these 36 nodes, we select 10 node-disjoint wireless links using the multi-graph reduction procedure as described in Section 4.3.8. We then proceed with the measurement steps for the 10 selected links as described above. An example of the selected links for the measurements of IRR is depicted in Figure 4.3.2. Only the links that have been selected for the measurements are displayed, each color of a link represents a different channel.

However, it is of interest to investigate the assumption that the selected links indeed meet the criteria of close spatial proximity resulting in interference effects when the links are activated simultaneously. For this reason, we validated the assumption in an experiment

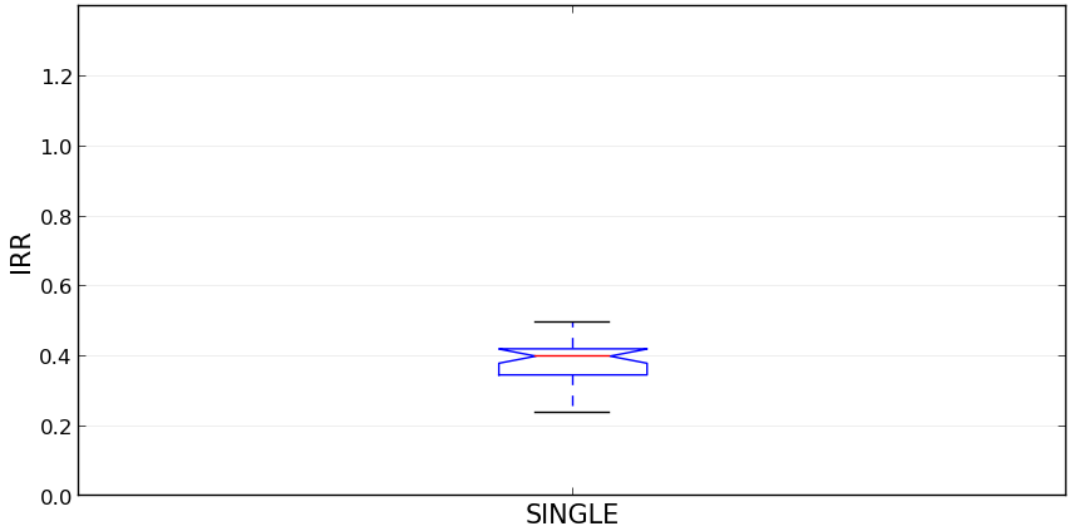


Figure 4.3.3.: Effects of inter-path interference in a single channel network. In a single channel network, the achievable throughput when activating all 10 links at the same time is only about 40% in the median compared to when the links are activated sequentially.

using one WNIC per node tuned to a global channel, resulting in a single channel network. We then measured the impact of inter-path interference as described above and replicated the experiment 40 times. The results for the IRR metric are depicted in Figure 4.3.3. The plot shows that the achievable throughput when activating all 10 links at the same time is more than halved compared to when the links are activated sequentially. Therefore, there is a high impact of interference effects and our selection procedure is indeed capable of meeting the requirement of close spatial proximity.

4.3.4 Saturation Throughput Ratio (STR)

The *saturation throughput* is described as the maximum load that the system can carry in stable conditions [110]. It can be determined by increasing the traffic load on the system until the limit is reached. This metric can be used to directly show an increase of the network capacity gained with a channel assignment algorithm.

For the measurement procedure, we define L as the set of all wireless links in the network. In the first iteration $i = 0$, we take a subset of k links $S_{i=0} \subseteq L$ with $|S_{i=0}| = k$ and the constraint that each node can only be adjacent to one link $l \in S_{i=0}$. We then measure the aggregate throughput of all links in $S_{i=0}$ simultaneously which is

$$SAT_{i=0} = \sum_{l \in S_{i=0}} t_l \tag{4.7}$$

For the next iteration $i = 1$, we add another k links with the same constraints and repeat the measurement. The procedure terminates if $SAT_{i+1} \leq SAT_i$ or S_i cannot be extended any further. In other words, the total aggregate throughput of the network will cease to

increase or no suitable links remain. The network is now saturated. Therefore we define the saturation throughput S as follows:

$$S = \max_{\forall i}(SAT_i) \tag{4.8}$$

In this way, we measure the absolute values for the saturation throughput and therefore, the results are specific to the testbed and lack interoperability. To solve this, we set the saturation throughput achieved by a channel assignment algorithm S_{CA} in relation to the saturation throughput of a single channel assignment S_{Single}. In other words, S_{CA} will be normalized using a single channel assignment as a baseline and expresses how much the saturation throughput has increased over the single channel network.

As performance metric, we use the *saturation throughput ratio* (STR) as follows

$$STR = \frac{S_{CA}}{S_{Single}} \tag{4.9}$$

A value of STR greater than one indicates an increase in the network capacity due to the channel assignment, i.e., STR = 2 means that the network capacity has been doubled compared to the single channel network. A value of STR close to 1 indicates that the algorithm has almost no impact on the network capacity.

Scenario

In order to measure STR on the DES-Testbed, we chose $k = 10$ for the step size. Thus, in the first step we select 10 wireless links and measure the aggregate throughput of these links. For the next iteration we add another 10 links and repeat this procedure until the network is saturated or no more additional node-disjoint links can be found using the multi-graph reduction procedure as described in Section 4.3.8. For the algorithms being compared, we take the maximum number of links that have been found with all algorithms in the particular iteration. An example of the maximum number of selected links is depicted in Figure 4.3.4. Only the links that have been selected for the measurements are displayed, each color of a link represents a different channel.

4.3.5 External Interference Ratio (EXR)

For channel assignment algorithms that consider external interference, a performance metric is required to estimate the performance of the algorithms in environments with external interference. We define the performance metric as follows. First, we select a subset I of all network nodes N to act as interferers. With the number of interferers $i = |I|$, we are also able to control the level of interference created in the network environment by the interfering nodes. The activity of the interferers can vary from jamming one or more communication channels to using more WLAN specific bursty traffic patterns. The remaining nodes $M = N \setminus I$ run the channel assignment algorithm.

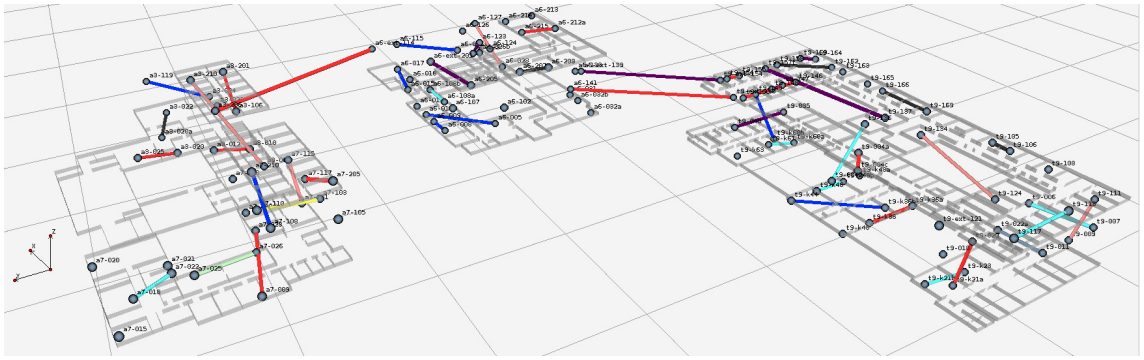


Figure 4.3.4.: Example of the link selection for the STR measurements. The example displays all found node-disjoint links after the channel assignment procedure.

For the performance evaluation, we select all links L in the network after the channel assignment with the additional constraint that every network node can only be adjacent to one link $l \in L$. Then, we measure the aggregate throughput of all links in L when they are activated simultaneously, which is

$$T_{noif} = \sum_{l \in L} t_l^{noif} \tag{4.10}$$

We repeat this measurement, but this time, the interferers I are active throughout the measurement. The aggregated throughput of the second measurements is defined as

$$T_{if} = \sum_{l \in L} t_l^{if} \tag{4.11}$$

The *external interference ratio* (EXR) metric is then defined as the ratio between the aggregated throughput while the interferers are active and when they are not, which is

$$EXR = \frac{T_{if}}{T_{noif}} \tag{4.12}$$

A value for EXR close to 1 means that the aggregate throughput is not reduced when the external interferers are active compared to when they are not. In this case, the algorithm avoids heavily congested channels and thus is not affected by the transmissions of the interferers. A lower value for EXR means that the transmissions of the interferers have decreased the aggregated throughput.

Scenario

In a first step, we randomly select the subset I of all network nodes to act as interferers with $I \subset N$, $|I| = i$. While the interferers are activated, the remaining nodes $M = N \setminus I$ run the channel assignment algorithm. After the channel assignment procedure, we select the maximum number of node-disjoint links considering the nodes in M using the multi-graph reduction procedure described in Section 4.3.8. With the selected links, we start the

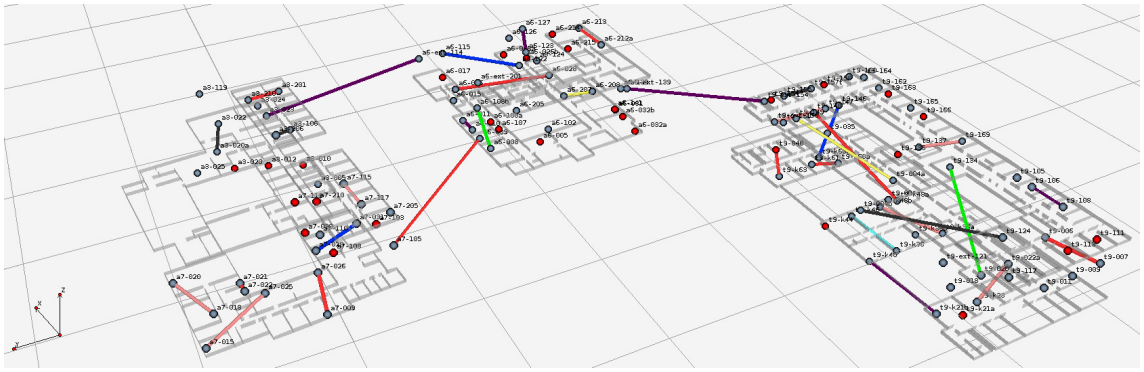


Figure 4.3.5.: Example of the link selection and interferers for the EXR measurements. The example displays the randomly chosen interferers depicted in red and the selected node-disjoint links after the channel assignment procedure.

measurement procedure of EXR as described above.

An example of the selected interferers and the selected links for the measurements is depicted in Figure 4.3.5. The interferers are depicted as red nodes, each color of a link represents a different channel.

4.3.6 Protocol Overhead (PO)

The overhead of a protocol is the additional amount of control messages the channel assignment approach imposes on the network. The control messages are exchanged between nodes, either on a regular basis or on demand. They are used for example to negotiate channel switches or to inform the neighboring nodes of changes in the channel assignment. As those control messages use the same communication paths as application data, they consume resources that would have been otherwise available to the latter. Consequently, the protocol overhead negatively affects the goodput of the network. Thus it is desirable, that channel assignment approaches have a minimum protocol overhead.

A metric for protocol overhead has to consider the amount of control messages and the amount of nodes involved in exchanging those messages. As performance metric, we define the *protocol overhead* (PO) as the mean number of messages sent by each node with

$$PO = \frac{1}{|N|} \cdot \sum_{u \in N} m_u \tag{4.13}$$

where N denotes the network nodes and m_u is the number of protocol messages sent by node u.

Scenario

The communication module of the DES-Chan framework as described in Section 6.2.4, allows to log all sent control messages during the execution of the algorithm. Thus, we can analyze the log file and determine the number of sent control messages per network node.

Metric	Motivation	Values	Interpretation
IAR	Measure the reduction of intra-path interference effects.	$[0, 1]$	The higher, the better - meaning a greater reduction of interference effects caused by intra-path interference.
IRR	Measure the reduction of inter-path interference effects.	$[0, 1]$	The higher, the better - meaning a greater reduction of interference effects caused by intra-path interference.
EXR	Measure the reduction of interference effects caused by external interferers.	$[0, 1]$	The higher, the better - meaning the throughput is less affected by the radio activity of external interferers.
STR	Measure the network capacity compared to the single channel network.	$[0, \infty]$	The higher, the better - meaning a higher network capacity has been achieved.
PO	Protocol overhead imposed on the network by the algorithm.	$[0, \infty]$	The lower, the better - meaning that less control messages have been sent per node.

Table 4.3.1.: Summary of the performance metrics of the channel assignment benchmark.

4.3.7 Summary of performance metrics

All developed performance metrics for the channel assignment benchmark are summarized in Table 4.3.1. The summary includes the motivation, possible values, and interpretation of each metric. Altogether, all sources of interference as described in Section 2.2 are considered.

4.3.8 Implementation

CAB has been implemented primarily for the performance evaluation of channel assignment algorithms in the DES-Testbed. This section describes the architecture, workflow, and challenges of the implementation. The core of CAB is a controller script written in Python. The controller runs on the DES-Portal, the testbed server which has a wired connection to all mesh routers. Additional external tools have been used, such as traffic generators for throughput measurements, *parallel-ssh*[1] for remote command execution, and the ETX daemon of the DES-Chan framework to retrieve the network topology.

Controller and workflow

The controller takes care of all required steps to execute the benchmark on the DES-Testbed. This comprises executing the channel assignment algorithms and the scenarios to measure the performance metrics, gathering and processing log files to extract results, and

[1] http://www.theether.org

Algorithm 1 CAB workflow

```
1: for rep in replications do
2:     for alg in algorithms do
3:         network.init()
4:         alg.execute()
5:         for sce in scenarios do
6:             update_topology()
7:             sce.start()
8:         end for
9:     end for
10: end for
```

a light-weight plotting engine based on *matplotlib*[2] to visualize the results. A configuration file allows to define the algorithms and scenarios that should be evaluated. Additional meta data, such as the number of replications, the available network nodes and their WNICs, and log file folders, are specified in the configuration file as well.

The simplified control flow for the channel assignment benchmark is given in Algorithm 1. Before the channel assignment algorithm is executed, all testbed nodes and the configuration of the wireless interfaces are reset. Then, the channel assignment algorithm is executed on the participating nodes. When all instances of the algorithm terminate with the final assignment on all nodes or a timeout has been reached, the network topology is updated. Finally, the measurements for the particular scenario are started.

Retrieving the network topology

Several of the described performance metrics are based on measuring the throughput of a subset of all wireless links in the network. The benchmark controller therefore requires knowledge about the network topology. For this reason, the controller can trigger an update of the network topology realized with an ETX daemon, which detects all links in the network and their quality using the ETX link metric [100]. Based on the network topology, links can be selected under the conditions of the particular performance metric.

Multi-graph reduction

After updating the network topology, several of the introduced performance metrics require the selection of a subset of all links for the following throughput measurements. However, the selection process of these links yields several pitfalls and therefore requires careful consideration. Due to the broadcast nature of the wireless medium, it is very likely that the resulting communication graph in a multi-radio network after the channel assignment is a multi-graph with multiple links between nodes. Since we introduced the constraint for several metrics that the selected links are all node-disjoint (each node is adjacent to exactly one or none of the selected links) at most one link between any pair of nodes can

[2] http://matplotlib.org

be selected. Selecting one of the links between two nodes randomly can lead to several problems due to the nature of many channel assignment algorithms.

Several studies in the domain of channel assignment have emphasized that appropriate metrics on the upper layers, such as routing metrics, are required to exploit the benefits of the spectral diversity. This leads to preferring links on a path with a higher spectral diversity than the shortest path between two network nodes. Several interference-aware routing metrics have been proposed that favor channel-diverse paths, such as WCETT [72], MCR [66], iAware [111], and EETT [112]. Additionally, several approaches exist for a combined solution of channel assignment and routing [113–115].

The studies have shown that it is important to consider the spectral diversity in the link selection procedure in order to assess the potential performance gain due to the channel assignment. This consideration is especially important for channel assignment algorithms that rely on a global common channel in order to preserve the network connectivity. Using such a global channel has the effect, that if two nodes reside in each others communication range, they can most likely communicate since both nodes have an interface tuned to the global channel. Additional links between these nodes may exist if on each node at least one additional radio is tuned to another mutual channel. However, the proportion of links in the network on the global channel compared to links using channels other than the common global channel is very high.

As an example, we investigate the results of the performance evaluation of the MICA and DGA channel assignment algorithms, which is described in detail in Chapter 7. The percentages of links tuned to the common global channel per algorithm are depicted in Figure 4.3.6. For the randomized channel assignment approach RAND, 86% percent of all links are operated on the common global channel, 80% for DGA, and 74% for MICA. In the single channel network (SINGLE), all links are tuned to the global common channel. Picking links randomly is obviously a bad idea, since we can expect that most selected links are using the global channel. Instead, an interference-aware mechanism has to be used similar to the mentioned interference-aware routing metrics. A simple adaption of the interference-aware solution is to prefer links in the selection procedure, that use a channel other than the global common channel. This is also emphasized since the communication over the global channel is usually only intended to be used as a last resort, meaning only for nodes, that are not connected otherwise. Therefore, for the selection process, links that are not tuned to the global channels are preferred.

Traffic generators

Throughput is one of the most used metrics to investigate the performance of wireless networks. In order to measure throughput, network traffic has to be injected into the network. This can either be done by replaying previously recorded network traffic, or traffic generators can be used. These generators can be used to inject artificial, customizable traffic patterns into the network.

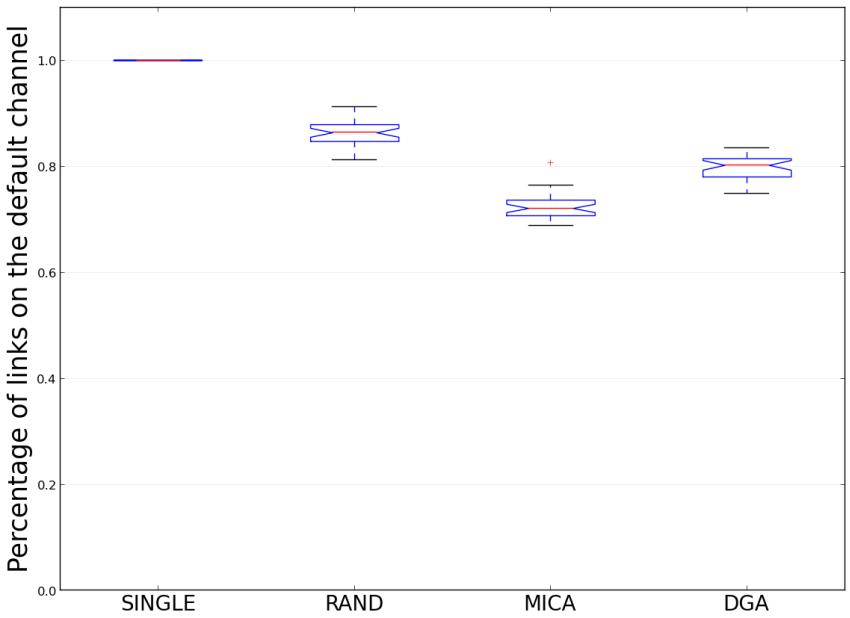


Figure 4.3.6.: Percentage of links tuned to the common global channel per algorithm.

Two popular traffic generators are `iperf` [116] and `nuttcp` [117]. Both tools are open source and allow to set several parameters, such as the transport protocol, message size, flow duration, and many more. `iperf` has certainly the larger user base and has been used to measure the performance of wired and wireless networks alike. `iperf` is based on a client-server approach and allows to measure the throughput between the client and server uni- and bidirectionally. Control message to signal the start and end of a traffic flow are exchanged over the same connection, meaning there is no separate control channel for these messages. This can lead to problems in wireless environments with high level of interference when control messages are lost due to message collisions or bottlenecks. `iperf` tries to solve this problem by using an ACK mechanism. When no ACK message has been received for a particular control message, this message is rescheduled up to 10 times.

`nuttcp` is a traffic generator based on `nttcp` and `ttcp`, the latter was developed already in 1984 [118]. The feature set of `nuttcp` is similar to `iperf`. Additionally, `nuttcp` allows to specify a different communication channel for control messages other than the connection on which the throughput is measured on. This is especially useful for network architectures where wireless routers are also equipped with a wired network interface. The control messages can then be exchanged over the more reliable wired interface, while the actual throughput measurement is performed on the wireless channel.

Most of the proposed performance metrics of CAB are based on throughput measurements in environments with a high level of interference. In order to measure the efficiency of channel assignment algorithms, several measurements rely on simultaneous throughput measurements across the whole network. This may lead to high packet loss rates. If control packets are affected by the packet loss, the measurements are not successful and no statistics of the transferred traffic can be made. For this reason, the idea to use a separate wired

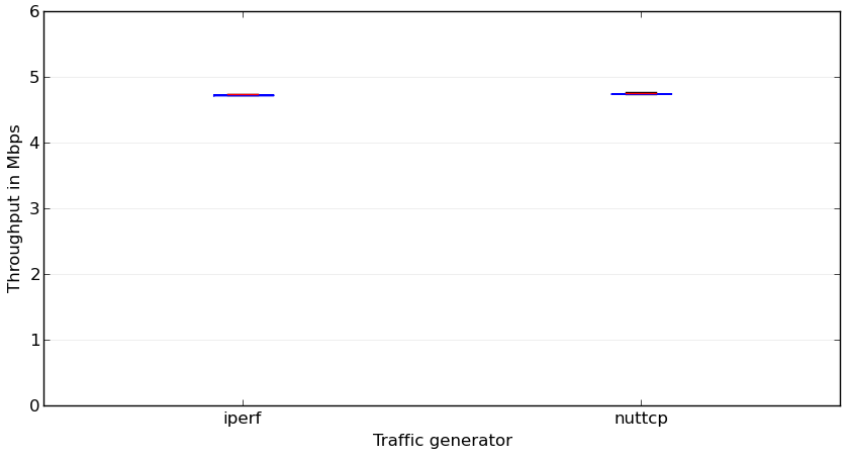


Figure 4.3.7.: Throughput results for a single link with `iperf` and `nuttcp`.

control channel sounds attractive. Therefore, we investigate the impact of unsuccessful measurements due to lost control messages and if using a separate control channel is able to reduce the number of unsuccessful measurements.

Experimental evaluation In a first experiment, we compare the throughput measurements of a single link in an interference-free environment (see Section 4.3.1). Both traffic generators are set to generate UDP traffic with a data-rate of 6 MBit/s and a datagram size of 1024 Byte for a flow duration of 30 seconds. The wireless network interfaces corresponding to the link were set to a channel in the 5 GHz spectrum and the data rate was set to 6 MBit/s as well. The results for 100 throughput measurements are depicted in Figure 4.3.7. The mean throughput over all measurements with `iperf` is 4.72 MBit/s compared to `nuttcp` with 4.74 MBit/s. All measurements have been successful in a way, that no control messages were reported as lost and for the measured throughput T holds $\mathsf{T} > 0.0$ MBit/s. The results are in the expected order and the boxplots show, there is hardly any deviation across the measurements.

As next, we investigate the performance of the traffic generators in an environment with a high level of interference. For this scenario, we use 126 nodes of the DES-Testbed and set up one wireless interface per node to a common global channel in the 5 GHz spectrum. After the network initialization, we select as many node-disjoint links as possible, meaning that each network node is at most incident to at most one of the selected links. We then measure the throughput simultaneously on all selected links. The scenario is adapted from the *saturation throughput ratio* (STR) performance metric as described above.

With 35 experiment replications, this results in a total of 4244 throughput measurements for both traffic generators. Statistics about the measurements considering lost control messages and successful measurements with $\mathsf{T} > 0.0$ MBit/s are given in Table 4.3.2. For the results, we calculate the percentage of measurements in which control packets have been reported as lost. Additionally, we classify all measurements that reported a throughput T with $\mathsf{T} > 0.0$ MBit/s as successful. Most notably, we observed that in about 15.7%

	Total flows	Successful flows	Lost control messages
iperf	4244	3674 (85.6%)	668 (15.7%)
nuttcp	4244	3897 (91.8%)	0

Table 4.3.2.: Results of experiments with the traffic generators `iperf` and `nuttcp`. The feature of `nuttcp` to use a different (wired) network interface for control message results in more successful measurements.

of the measurements with `iperf` that control messages have been lost. The statistic is based on the exception thrown at the client, stating, that ACK messages have not been received after 10 tries. The number of unsuccessful measurements as reported by the `iperf` server is a little lower (14.4%) since these reports are generated independently of the client receiving the final ACK at the end of the traffic flow. In contrary, no control messages have been reported as lost for the measurements with `nuttcp`. The percentage of successful measurements is also significantly higher with 91.8%.

In summary, the experimental study shows, that the feature of `nuttcp` to use a different communication channel for control messages is capable of increasing the amount of successful measurements. This is especially useful for a network architecture such as the DES-Testbed, in which each wireless mesh routers has a wired connection to the testbed server. Thus, the wired interface can always be used as control interface for the throughput measurements with `nuttcp`. Based on the results of this study, `nuttcp` has been chosen as default traffic generator for the channel assignment benchmark. However, `iperf` is still supported. The desired traffic generator can be set in the configuration file of the channel assignment benchmark.

Plotting engine

CAB provides a light-weight plotting engine based on the Python plotting library *matplotlib* in order to visualize the processed results. Per default, results are plotted as boxplots, using 95% confidence intervals, 25 and 75% quantiles, and whiskers.

Results

CAB has been used for the performance evaluation of the channel assignment algorithms described and developed in this dissertation. The results of studies on link-based and interface-based channel assignment are presented in Chapter 7, results of studies on external-interference aware channel assignment are given in Chapter 8.

CHAPTER 5

Channel occupancy interference model

The impact of interference on the performance of different protocols, such as channel assignment [41,119], load balancing [120], and routing [72,112] has been extensively studied. As a result, interference-aware approaches have been developed. The performance of these protocols depends heavily on the accuracy of the initial *interference estimation* for the network. It is therefore essential for the performance of these protocols to understand interference in order to derive models and procedures to accurately describe interference relationships between network nodes. In this chapter, a linear measurement-based approach for interference estimation in large-scale wireless mesh networks is presented. Based on this method for interference estimation, the *channel occupancy interference model* (COIM) is developed. The chapter concludes with the results of the validation of COIM in a study on the DES-Testbed. Excerpts of this chapter have been published in [50].

5.1 Interference estimation

Obtaining an accurate interference estimation for a wireless network deployment is not a trivial task. Due to the complexity of the radio signal propagation, initially simplified models based on heuristics have been proposed for the interference estimation. One of these is the *point interference model*, stating that two links interfere if they share a common endpoint [47]. A more sophisticated model defines for each node (or more precisely radio when nodes are equipped with multiple radios) a transmission radius r_t and an interference radius r_i, with $r_t < r_i$. A transmission is correctly received at node u, when only one node of all the nodes in the interference range of u transmits at a time. Models based on this assumption usually approximate the m-hop neighborhood of a node u as the *interference set* [48], which consists of all nodes whose transmissions may interfere at u. A typical value is $m = 2$, meaning the interference range of a node is twice its communication range. This model has been first described as *protocol model* [35].

With these simple models, the interference estimation can be calculated without much effort based on the local neighborhood information of a network node. However, several studies have shown, that these simplifications do not hold in reality. The predicted in-

terference relationships are not accurate for indoor network deployments [46, 49, 50], the interference estimation is either too optimistic or pessimistic.

For this reason, *measurement-based* methods have been proposed that promise a higher degree of accuracy. With pair-wise approaches, measurements are performed for each pair of network nodes resulting in a complexity of $\mathcal{O}(n^2)$ for a network with n nodes. One example for such an approach is the *broadcast interference ratio* (BIR) [46] which approximates the interference of unicast traffic using broadcast transmissions. The study also showed that the *carrier sensing* (CS) mechanism of IEEE 802.11 [121] is the main cause for the performance degradation, resulting in sender-side deferral of messages. Pair-wise measurements are also used in the creation of *interference maps* in [51].

Pair-wise approaches have been usually designed and evaluated in smaller testbeds or network deployments with about 10-30 nodes [46, 51]. Although they deliver accurate results in these networks, they do not scale well for larger networks, since the measurements are based on traffic flows with a fixed duration that have to be carried out sequentially to avoid side effects. For example, if all nodes in a network are tested pair-wise, a network with 100 nodes would require 4,950 measurements. With each measurement lasting 20 seconds, the overall measurement procedure, in which the network can not be used for other applications, would take more than a day. Therefore, these approaches are not feasible for large or dynamic networks, in which the network topology changes over time.

To address this issue, several approaches [73, 122–125] proposed interference models that are seeded with only $\mathcal{O}(n)$ measurements. The measurement procedure is based on monitoring the *received signal strength indicator* (RSSI) values and the *packet delivery ratio* (PDR). However, this way only nodes are captured that are in each others communication range, since RSSI values can only be read from received packets. Therefore, nodes which are not in the communication range but in the interference range, also referred to as remote interferes [51], are ignored. In [122], the measurements are used to seed a physical layer model for message deferral and corrupted messages that considers single interferers. The approach has been extended to consider multiple interferers in [123] and [125]. Based on the correlation of the PDR and the *signal interference to noise ratio* (SINR) in IEEE 802.11 networks, a physical model is proposed in [73].

5.2 Channel occupancy measurements

Our approach for interference estimation is based on measuring the *channel occupancy* (CO) using the *carrier sensing* (CS) mechanism of the wireless network interface while possible interfering nodes are utilizing the shared carrier. The channel occupancy (also referred to as channel load or channel busy fraction) is a value between 0 and 1, describing what fraction of a fixed interval I the medium was sensed busy at a particular node.

More formally, we define

- CO_i^k as channel occupancy measured at node k while node i is sending at maximum rate in interval I

- T_{CO} as threshold for the channel occupancy. If $CO_i^k > T_{CO}$ then the transmissions of node i have been sensed at node k

CO represents an $n \times n$ matrix for a network with n nodes and each value $CO_i^k \in [0, 1]$. A value of $CO_i^k = 0.5$ means, that at node k, the medium has been sensed busy for 50% of the measurement interval I in which node i was sending at maximum rate. We introduce the threshold T_{CO}, because real-world network deployments are usually set up in unshielded environments. The environment may generate dynamic noise which occasionally exceeds the carrier sensing threshold. With the threshold T_{CO}, we are able to discard a very low channel occupancy caused by the network environment. The procedure to obtain a reasonable value for T_{CO} is described in detail in the next section.

As described in Section 5.1, several approaches for interference estimation are based on measurements of the RSSI and PDR of each node's transmission, making them only capable to detect nodes in the communication range. With our approach we measure the channel occupancy instead, which has the advantage that also nodes in the carrier sense range are captured. However, measuring the channel occupancy poses new challenges, because it requires the availability of an unused radio channel in the network deployment environment. Otherwise, it is impossible with carrier sensing alone to distinguish between channel occupancy caused by the activity of another network node, by the environment, or external networks and devices. For instance, the DES-Testbed is deployed mostly in office buildings, several WLANs are co-located that provide Internet access to students and the research staff. However, these WLANs operate mainly in the 2.4 GHz frequency band, but there is hardly any activity in the 5 GHz band, which can be used for the channel occupancy measurements.

5.2.1 Implementation

With commodity IEEE 802.11 hardware, we can measure the channel occupancy by directly accessing the hardware registers for carrier sensing of the *wireless network interface card* (WNIC). This way, we can monitor the activity on the medium for a specified interval and calculate the corresponding channel occupancy accordingly. The measurement procedure for CO in a network with nodes V is as follows. Node i broadcasts at maximum rate while all other nodes $V \setminus \{i\}$ measure the channel occupancy. All nodes broadcast one at a time, which results in a complexity of $\mathcal{O}(n)$ measurements.

In order to identify an unused channel for the measurements and to obtain a reasonable value for the threshold T_{CO}, we execute the following *sanity check* procedure. On all channels of the 5 GHz frequency band, we measure the channel occupancy on all nodes simultaneously while our network is completely silent, i.e., no traffic is generated. The results showed, that on the channels $\{36, 40, 48, 52, 56, 60, 64\}$, the measured channel occupancy varied between 0-0.7% for all nodes. This is low enough to qualify as unused, empty channels and allows us to choose a reasonable value for T_{CO}. We set $T_{CO} = 0.01$, meaning that the channel occupancy below 1% is considered as noise generated by the environment.

5.2.2 Experiment Design

For the measurement of the channel occupancy values in the DES-Testbed, we used one Atheros-based WNIC on each of the 128 network nodes. The data-rate of the WNIC is set to 6 MBit/s. On its turn, the sending node transmitted at full rate for 8 seconds (that is 6 MBit/s corresponding to the settings of the WNIC). All other nodes measured the channel occupancy with 10 samples, each sample capturing the channel busy time for an interval of 0.5 seconds. From these samples, we calculate the matrices for the *mean* sensed channel occupancy as CO_{Mean} and the *maximum* as CO_{Max}. With these settings, one execution of the experiment on the DES-Testbed with 128 network nodes lasted about 20 minutes. We run 50 replications of this experiment at night time resulting in a total runtime of roughly 17 hours.

5.2.3 Evaluation

As expected, the fraction of results, in which the sending node has not been sensed by the listening node is the highest. In average over the 50 experiment runs, 90% of the 16,384 entries of both matrices CO_{Mean} and CO_{Max} are below the threshold T_{CO}. This is as expected, since the DES-Testbed spans a large area comprising 3 buildings, so many nodes are located relatively far away of each other. The distribution of the values in CO_{Mean} and CO_{Max} for the remaining 10% (about 1,450 entries per experiment), in which the threshold T_{CO} was exceeded, are depicted in Figure 5.2.1.

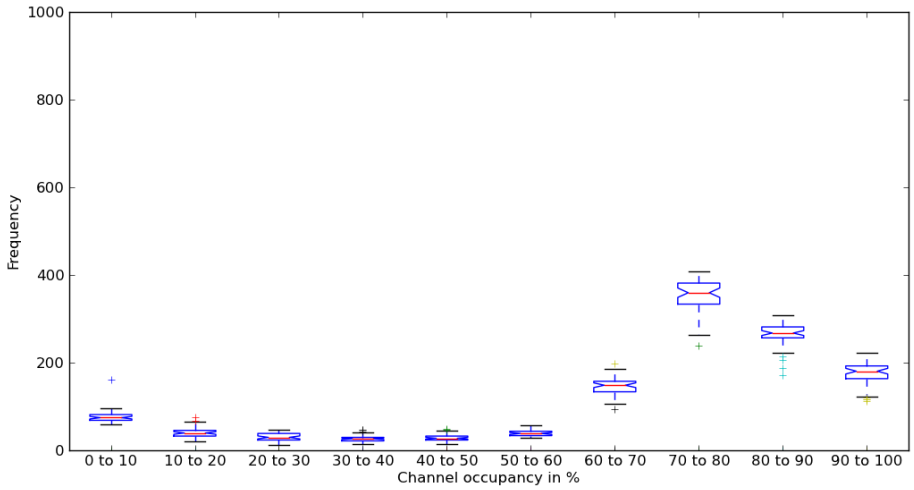
For the values in CO_{Max}, we can observe an almost binary distribution, in which the vast majority of the entries are greater than 90%. For the values in CO_{Mean}, we observe a higher number of values between 60 and 100%. Further investigation revealed that there is a correlation between the values in CO_{Mean}, which are greater than 60% and the values in CO_{Max} greater than 90%. Considering the median for each sender-listener combination (i, k) in CO_{Mean} and CO_{Max} over the 50 replications, we can derive that for about 90% of the entries it holds

$$mean(CO_i^k) > 0.6 \Rightarrow max(CO_i^k) > 0.9 \tag{5.1}$$

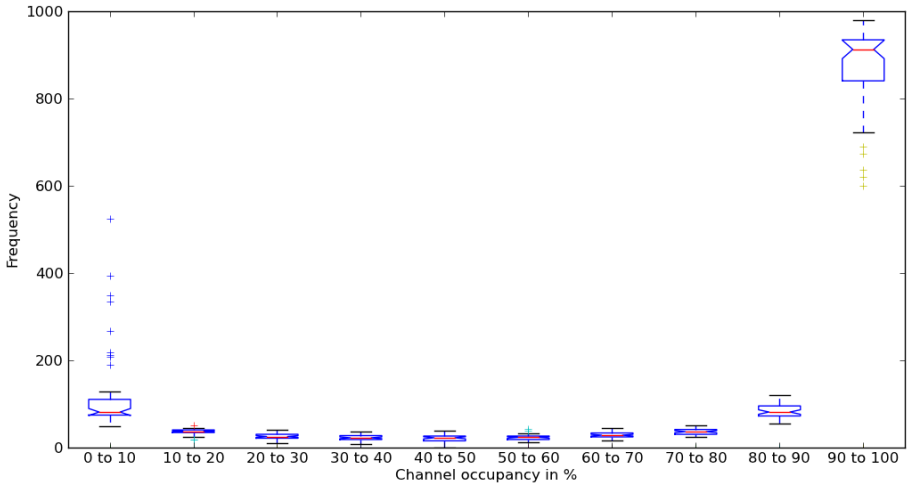
Taking a closer look at the values of CO_{Mean}, we calculate the standard deviation of the measurements as depicted in Figure 5.2.2. The plot shows, that there are many entries with a small standard deviation for $CO_i^k < 0.1$ and for $CO_i^k > 0.9$. However, for values with $0.1 < CO_i^k < 0.9$, the standard deviation of the measurements is higher.

This observation allows us to classify the sender-listener combinations (i, k) of the experiment. Using Equation (5.1), we can differentiate between three classes:

$$(i,k) \in \begin{cases} I_S & \text{if } mean(CO_i^k) > 0.6 \wedge max(CO_i^k) > 0.9 \\ I_W & \text{if } max(CO_i^k) > T_{CO} \wedge (i,k) \notin I_S \\ I_N & \text{if } max(CO_i^k) \leq T_{CO} \end{cases}$$



(a) Distribution of the entries with $CO_{Mean} > T_{CO}$.



(b) Distribution of the entries with $CO_{Max} > T_{CO}$.

Figure 5.2.1.: Distribution of values in CO_{Mean} and CO_{Max} which are greater than the threshold T_{CO}. While there is almost a binary distribution for the values in CO_{Max}, most values in CO_{Mean} are scattered between 60 and 100%.

In other words, class I_S comprises all combinations (i, k) in which node i is a *strong interferer* for node k, I_W comprises the combinations in which node i is a *weak interferer* for node k, and I_N comprises all combinations in which node i is considered not an interferer for node k.

It is important to note that I_N is the largest of these sets with 90% of the (i, k) combinations, I_S comprises 9%, and I_W only about 1%. We will use this observation for the development of an interference model in Section 5.3.

5.2.4 Correlation to network topology

As next, we investigate the results for the correlation between the measured channel occupancy and the network topology based on the communication graph. We obtained the

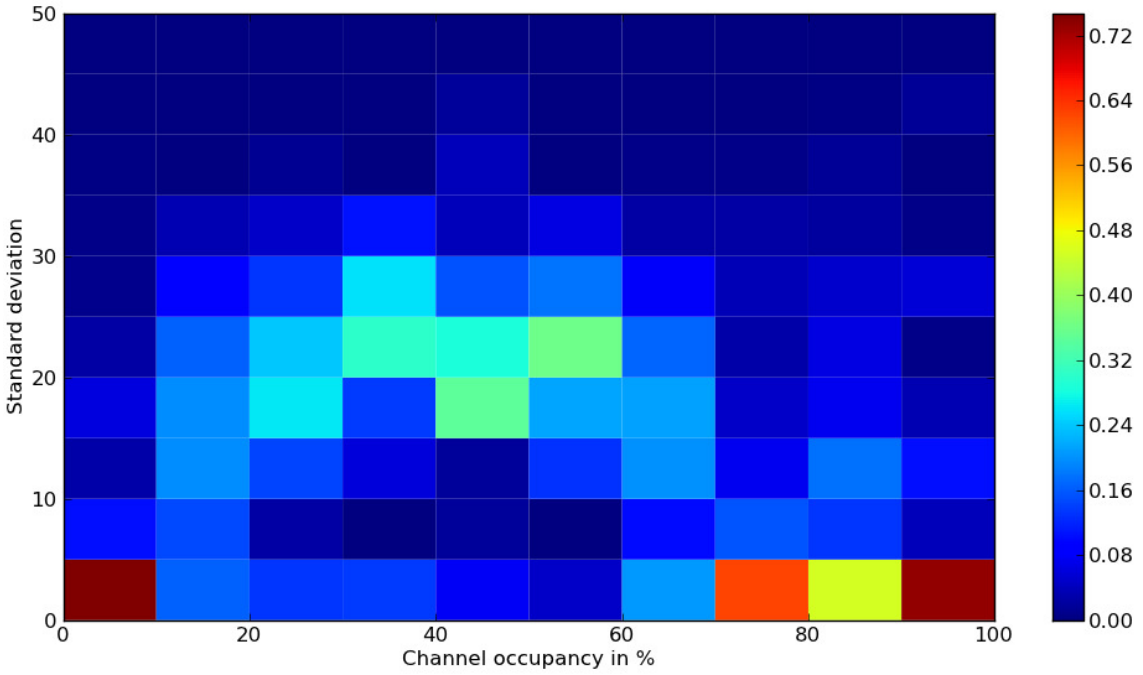


Figure 5.2.2.: Standard deviation of the values in CO_{Mean} over 50 experiment replications.

network topology of the DES-Testbed using a neighbor discovery service which delivered all communication links between the network nodes and their corresponding link quality using the *expected transmission count* (ETX) metric [100]. For the evaluation, we constructed the communication graph with all detected links regardless of the link quality, which means that also very lossy links are included.

The node degree based on the communication graph (the amount of 1-hop neighbors for each node), in relation to the node degree based on the channel occupancy measurements (nodes detected by carrier sensing), is depicted in Figure 5.2.3. As can be seen, for all nodes, the amount of 1-hop neighbors in the communication graph is always less or equal to the number of nodes detected by carrier sensing. The mean node degree over all nodes derived from the communication graph is 5.40 compared to 10.01 for the carrier sensing based node degree. This is an indication, that the proposed method is indeed capable of detecting nodes, that reside in the carrier sensing range of a node, but not in its communication range. It also shows, that the number of nodes outside the communication range but inside the carrier sense range is high, and therefore these remote interferers should not be ignored for the interference estimation.

The CDF for the node degree based on the communication graph and the node degree based on the carrier sensing are depicted in Figure 5.2.4a and Figure 5.2.4b. The comparison in Figure 5.2.4c shows the correlation between the two measurements. The fitted line shows, that we can predict the number of possible interferers based on the 1-hop neighbors quite well. This shows, that there is a correlation between the results of the channel occupancy measurements and the node degree based on the communication graph.

As next, we take a closer look on the relation between the carrier sensing results of a

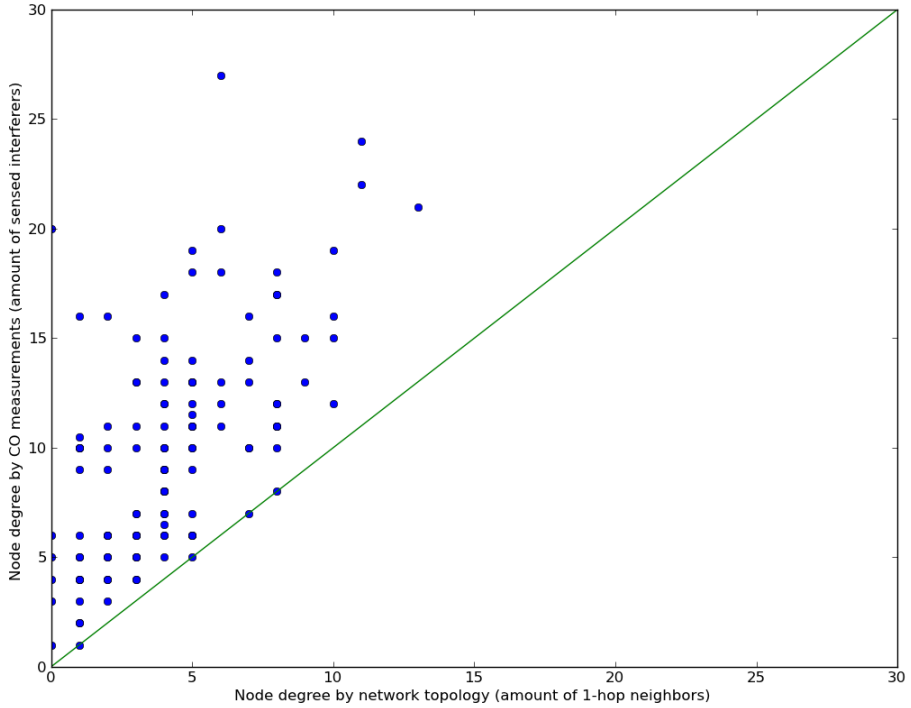


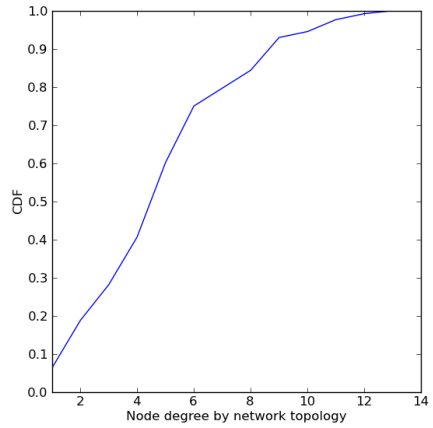
Figure 5.2.3.: Correlation of the node degree derived from the network topology and from the channel occupancy measurements by carrier sensing.

sender-listener combination (i, k) and the distance in hops between the two nodes $d_{i,k}$ in regard to the network topology. For each node we determined all 1-hop neighbors, and checked how many of these neighbors have been successfully detected with carrier sensing, meaning that $CO_i^k > T_{CO}$. We then did the same for the 2-hop, 3-hop, .., 6-hop neighbors. The results are depicted in Figure 5.2.5. As expected, over all experiments, almost all 1-hop neighbors have also been detected by the carrier sensing. For the 2-hop neighborhood, only about 40% of the nodes have been detected by carrier sensing. This is an interesting observation, because according to our assumption, that means that about 60% of the 2-hop neighbors are not interferers. This corresponds to other validation studies that found that the 2-hop interference model is too pessimistic for real network deployments [46, 49]. Additionally, our results also show, that the 2-hop interference model is also too optimistic in certain cases, since still 10% of the 3-hop neighbors are still detected by carrier sensing.

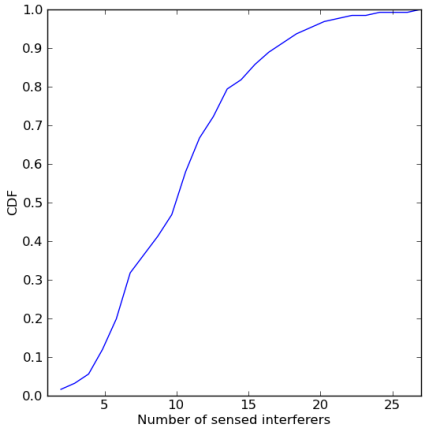
The results show once more the complexity of signal propagation modeling in indoor environments. This task is especially hard when additional dynamics such as moving people, open and closed doors and windows have to be taken into account as well. This underlines that measurement-based approaches are required in order to achieve more accurate results.

5.2.5 Measurements over time

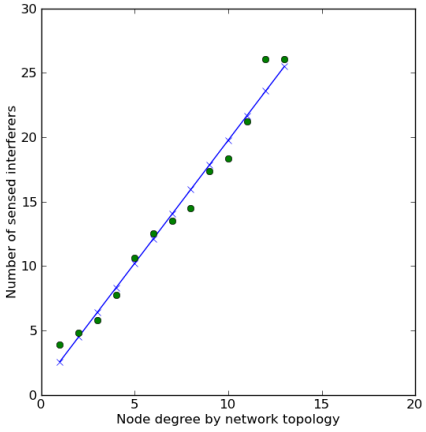
The DES-Testbed as the experimentation environment for the channel occupancy measurements is a static wireless mesh network, meaning that all nodes reside in a fixed location. However, the environment of the DES-Testbed features a certain dynamic, since the nodes



(a) CDF of node degree by network topology



(b) CDF of node degree by carrier sensing



(c) Comparison between node degree by network topology and carrier sensing

Figure 5.2.4.: Comparison of the CDF of the node degree based on the communication graph and the number of nodes detected by carrier sensing.

are deployed in offices, seminar rooms, and lecture halls across different buildings. During day time, students might roam the buildings, research staff members might move nodes to slightly different locations or by accident move the antennas a little bit.

Therefore, it is interesting to measure the channel occupancy at different times and compare the measurements in order to assess to what degree the interference patterns change over time. We chose three scenarios to compare the results of the measurements. First, we compare a measurement performed during day time with one performed at night. Second, we compare two measurement that are three days apart at night time, and finally, two measurements 6 week apart.

The results of the error rate for each scenario is summarized in Table 5.2.1. The error rate is defined as the percentage of entries in CO_{Mean} (CO_{Max} respectively) which are below the threshold T_{CO} in one measurement and above in the other or vise versa. In other words, in one measurement the transmissions of a node have been sensed at another node

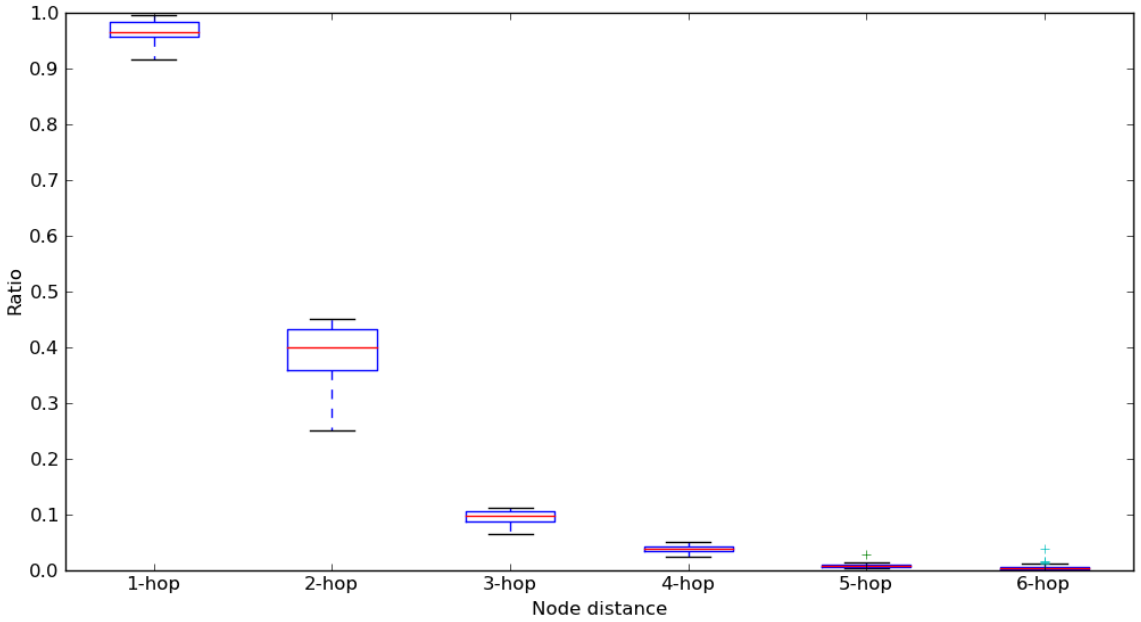


Figure 5.2.5.: Ratio of detected nodes via carrier sensing regarding the distance between the sender and listener in hops based on the communication graph.

Time span between measurement	Error rate CO_{Mean}	Error rate CO_{Max}
Day vs night	1.4%	2.1%
3 days apart	1.5%	2.1%
6 weeks apart	4.3%	6.6%

Table 5.2.1.: Error rates for CO_{Mean} and CO_{Max} for measurements over time.

and in the other the transmissions have not been sensed. For the first scenario, we observe an error rate of 1.4% for CO_{Mean} and 2.1% for CO_{Max}. We observe almost the same error rate for the measurements performed three days apart, both of them at night time. The error rates for the measurements six weeks apart are significantly higher, with 4.3% for CO_{Mean} and 6.6% for CO_{Max}.

The results show that for a static network in a mostly static environment, the patterns for the channel occupancy measurements are different at different times and the difference increases with the time span between them. Therefore, the measurements have to be repeated periodically, stressing the importance of an efficient measurement procedure (i. e., in linear runtime).

5.3 Interference model

Based on the analysis of the channel occupancy measurements, we define the *Channel Occupancy Interference Model* (COIM). First, we define the following

- $l_{k,j}$ direct link between nodes k and j

- $tp^{k,j}$ achievable throughput from node k to node j
- $tp_i^{k,j}$ throughput from node k to node j with interfering node i sending simultaneously at maximum rate

COIM is based on the following two assumptions:

$$1.\ CO_i^k < T_{CO} \wedge CO_i^j < T_{CO} \Rightarrow tp_i^{k,j} = tp^{k,j} \tag{5.2}$$

In other words, if CO_i^k and CO_i^j *do not exceed* the threshold T_{CO}, then the simultaneous transmissions of node i do *not* affect the capacity of the link $l_{k,j}$. In this case, node i is not an interferer for the link $l_{k,j}$. We will refer to this case as *NO-IF* throughout the remainder of this chapter.

The second assumption complements the model with

$$2.\ CO_i^k \geq T_{CO} \vee CO_i^j \geq T_{CO} \Rightarrow tp_i^{k,j} < tp^{k,j} \tag{5.3}$$

Meaning, that if CO_i^k or CO_i^j *exceed* the threshold T_{CO}, we expect a decrease of the capacity of the link $l_{k,j}$. In this case, node i is an interferer for link $l_{k,j}$. We will refer to this case as *IF*.

With these two assumptions, we define the binary interference model COIM for a link $l_{k,j}$ and a possible interfering node i as follows.

$$COIM(l_{k,j}, i) = \begin{cases} 0 & \text{if case NO-IF in Equation (5.2) holds} \\ 1 & \text{if case IF in Equation (5.3) holds} \end{cases}$$

In other words, if $COIM(l_{k,j}, i) = 1$, then the node i is an interferer for the link $l_{k,j}$, meaning that the transmissions of node i affect the capacity of the link $l_{k,j}$. On the contrary, if $COIM(l_{k,j}, i) = 0$, then the node i is not an interferer for the link $l_{k,j}$. The choice for a binary model lies in the analysis of the results of measuring the channel occupancy, which showed, that 99% of the sender-listener combinations (i, k) are either part of the class of strong interferers I_S or part of the class of non-interferers I_N. Additionally, as our validation experiments for COIM show, interference effects are of a very dynamic nature in real network deployments, thus making it practically very hard to predict the effects of an interfering transmission on a fine-grained scale.

5.4 Validation

We run the following experiment to validate the first part NO-IF of COIM as described in Equation (5.2). We randomly selected 28 sample combinations, each existing of a link $l_{k,j}$ and an interfering node i. For each sample $(l_{k,j}, i)$, according to the assumption in NO-IF, for the corresponding values holds $CO_i^k, CO_i^j < T_{CO}$. Using our classification, this is equivalent to $(i, k), (i, j) \in I_N$. Additionally, we introduce a further constraint regarding

the distance of the nodes k and i, in order to get more meaningful results. For each sample combination, we first randomly select the interfering node i, and then select the link $l_{k,j}$ of all links of which $CO_i^k, CO_i^j < T_{CO}$ and the distance $d_{i,k}$ between nodes i and k in hops based on the communication graph is minimal. More formally, for each sample $(l_{k,j}, i)$ holds

$$CO_i^k, CO_i^j < T_{CO} \wedge \min_{k \in V \setminus \{i\}} d_{i,k} \tag{5.4}$$

With this constraint, node k is the closest to node i in regard of the distance in the communication graph for which node i is not an interferer. This way, we avoid samples, in which the adjacent nodes of $l_{k,j}$ are relatively far away of i regarding the network graph and interference effects are unlikely to appear. As a result of applying this additional constraint, for all sample combinations the distance is $d_{i,k} = 2$, meaning that the interferer i resides in the 2-hop neighborhood of the node k. Thus, the results of this experiment will give us also insights about the performance of the 2-hop interference model, since according to the 2-hop model, all sample combinations should suffer from interference.

For the validation of the second part IF of COIM as described in Equation (5.3) we randomly select 28 sample combination $(l_{k,j}, i)$ in which we expect that node i is an interferer to the link $l_{k,j}$. Therefore, for each sample $(l_{k,j}, i)$, according to the assumption in IF, for the corresponding channel occupancy values holds $CO_i^k \geq T_{CO}$.

For both experiments, the following sequential steps are executed for each sample combination.

1. Measure the throughput $tp^{k,j}$ for a fixed interval of 10 seconds. We use `iperf` to generate an UDP flow with a data-rate of 6 MBit/s.

2. Start sending at maximum rate at node i and measure the throughput $tp_i^{k,j}$ with the same parameters as in step 1. We use a UDP traffic generator based on Python Twisted [126] for the transmissions of node i.

Both experiments were repeated 35 times for each sample combination, leading to a total runtime of 16 hours. For each experiment replication, we are interested in the ratio of the achievable throughput when the interferer i is active simultaneously with the flow on link $L_{k,j}$ and when it is not. More formally, that is expressed by

$$RTP_i^{k,j} = \frac{tp_i^{k,j}}{tp^{k,j}} \tag{5.5}$$

If the transmissions of node i do not affect the capacity of $l_{k,j}$, the measured throughputs should be the same and $RTP_i^{k,j}$ will be close to 1. On the contrary, we expect values for $RTP_i^{k,j}$ of less than 1, if the node i is an interferer for the link $l_{k,j}$. Since link qualities and signal propagation are known to be dynamic over time in unshielded testbed environments, we use the following threshold to quantify interference. We define that node i is an interferer to $l_{k,j}$ if $RTP_i^{k,j} < 0.85$. On the contrary, node i *is not* an interferer to $l_{k,j}$ if $RTP_i^{k,j} \geq 0.85$.

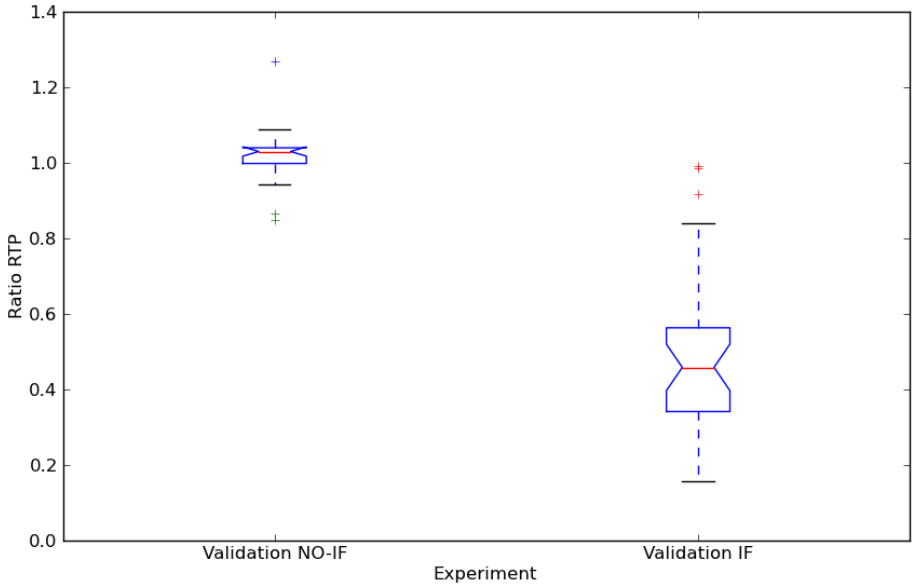
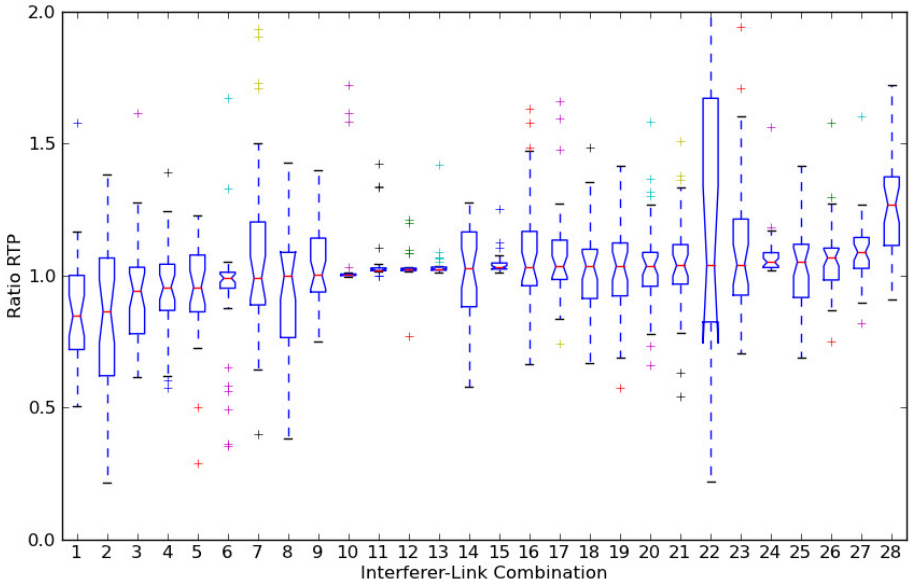


Figure 5.4.1.: Aggregated results for the experiments to validate COIM. For the validation of NO-IF the results over all sample combinations are in the median close to 1, whereas the results for IF are close to 0.4.

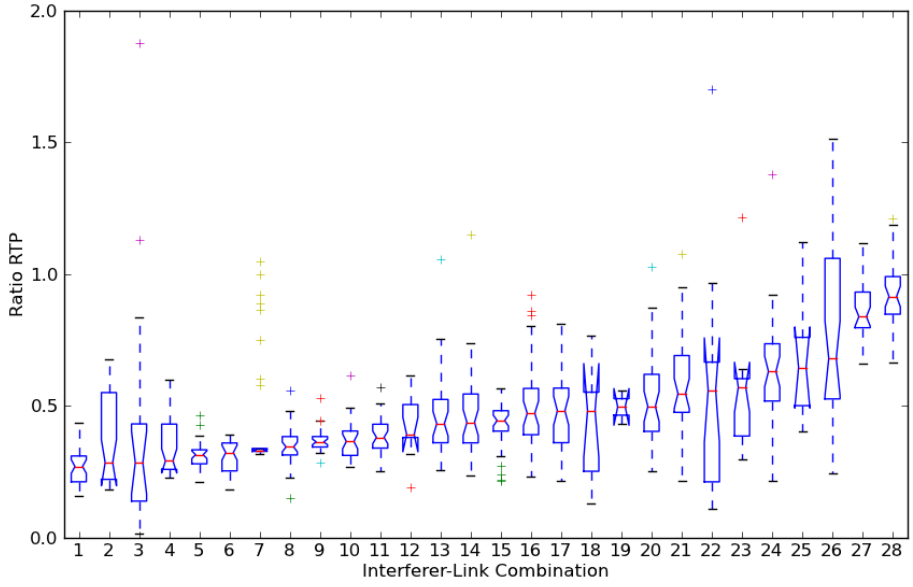
The aggregated results over the two experiments for the validation of COIM are shown in Figure 5.4.1. It can be seen that for NO-IF the results over all sample combinations are in the median close to 1 and the low height of the boxplot shows that the deviation is small. The results for IF are close to 0.4 in the median, which means, that the impact of interference on a particular link can severely affect the capacity of that particular link. For the median, the expected throughput is less then half of what can be achieved in the absence of the interfering node.

The results for each sample combination for the validation of NO-IF are depicted in Figure 5.4.2a and the results for IF in Figure 5.4.2b. For both graphs, it can be seen by the hight of the boxplots that the deviation over the 35 replications for each sample is quite high which reflects the dynamic link and signal propagation characteristics in real unshielded networks. In the case of NO-IF, the median of $RTP_i^{k,j} \geq 0.85$ for all sample combinations, meaning that we successfully validated this part of COIM. This allows us also to estimate the accuracy of the 2-hop interference model, since the distance between the possible interferer i and node k is $d_{i,k} = 2$. According to the 2-hop interference model, in all sample combinations interference effects should have been appeared. Since this is not the case, the 2-hop model is obviously too pessimistic in our testbed, and most likely for other network deployments as well.

For IF, the median of $RTP_i^{k,j} < 0.85$ for all sample combinations but one, meaning that the interference effects of the simultaneous transmissions of node i do have an effect on the capacity of the link $l_{k,j}$ for all but one sample combinations. The results show, that COIM is able to predict if a node i is an interferer for a link $l_{k,j}$ and will decrease the capacity for that particular link. However, we are not able to quantify the impact of the



(a) Validation of NO-IF. In samples with a resulting $RTP_i^{k,j} < 0.85$ for the median, node i is not interfering with the link $l_{k,j}$.



(b) Validation of IF. In samples with a resulting $RTP_i^{k,j} \geq 0.85$ for the median, node i is not interfering with the link $l_{k,j}$.

Figure 5.4.2.: Results for the validation of COIM. The results for each sample combination and the corresponding throughput ratio $RTP_i^{k,j}$ are plotted.

	COIM Prediction	
	NO-IF	IF
Observed Interference	0	28
Observed No-Interference	27	1

Table 5.4.1.: Prediction table for the validation of COIM.

simultaneous transmissions of the interfering node. As seen in the results, the medians of $RTP_i^{k,j}$ are quite evenly distributed between 0.26 and 0.91. Both results according to the predictions of COIM are summarized in Table 5.4.1.

5.5 Relation to channel assignment

The study has shown, that COIM provides a high degree of accuracy for the interference estimation. In a next step, we will compare the impact of channel assignment on the performance in regard of the utilized interference model. Since a measurement-based approach provides a more accurate interference estimation than simple heuristics, we expect that the performance of channel assignment algorithms can be increased by using more accurate interference models. By executing the same algorithm with different interference models, we are able to quantify the performance gain that can be credited to a more accurate models.

CHAPTER 6

A framework for distributed channel assignment algorithms

An important part of this dissertation is the evaluation of channel assignment algorithms in a real network environment. For this reason, implementation candidates for the experimental environment in question, the DES-Testbed, have to be developed. This chapter introduces DES-Chan, a software framework that provides common services and data structures for channel assignment algorithms to support the researcher in the development process. Excerpts of this chapter have been published in [127, 128].

6.1 Motivation

Several distributed channel assignment algorithms have been evaluated in small wireless testbeds [36,48,65,66]. However, an experimental study is lacking that compares the performance of multiple different algorithms in a large-scale wireless testbed. One of the reasons is the huge effort it takes to develop an implementation candidate of such an algorithm for real network environments. The development process yields several challenges and pitfalls: The researcher has to deal with operating system specifics, drivers for the wireless interfaces, and the capabilities and limitations of the particular hardware. Additionally, common operating systems such as Linux and Windows do not support multi-channel protocols out of the box. Furthermore, if more than one particular algorithm should be studied, common problems and services have to be addressed multiple times. Such a common service for example is the *interface management*, which is needed in order to carry out channel switches and to configure the wireless network interfaces. Additionally, distributed channel assignment algorithms usually require information about the local network topology and the channel selection therein. This task can be solved by providing a *neighborhood discovery* service that allows to retrieve all direct neighbors. By providing such services to the developer, a research framework for channel assignment algorithms can significantly speed up the development process.

Based on these observations, we developed the *Distributed, Embedded Systems - channel assignment framework* (DES-Chan) with the following key features:

- *Abstraction layer:* DES-Chan provides an abstraction layer in order to hide operating system details from the developer, such as memory management and handling the wireless interfaces. Another important part of distributed channel assignment algorithms are communication protocols to exchange information about the local channel selection and to negotiate channel switches. A framework can provide an API for these tasks and can handle sending and reception of messages. Thus, the overall development effort can be significantly reduced for the researcher.

- *Service library:* Despite the diverse strategies of channel assignment algorithms, there are several key services that are required by most algorithms. Therefore, the framework should provide a set of such common services as derived from the analysis of distributed channel algorithms discussed in Section 2.5. By providing such common services, the development effort can be significantly reduced and the framework enables rapid prototyping of implementation candidates for new algorithms.

- *Modular architecture:* A modular architecture is the foundation for an extensible framework. The framework should not limit the developer to the already provided services, since further channel assignment algorithms may require additional services. By encapsulating the functionality in modules, further services can be added on demand to the framework.

DES-Chan offers two main contributions to the research community. First, the effort to develop implementation candidates for channel assignment algorithms is significantly reduced. Due to the abstraction layer and the provided services, ideas for channel assignment strategies can be quickly implemented as a proof of concept in testbed environments. Thus, a wide range of different algorithms can be implemented and validated in a real network environment. Second, with the reduced development effort, diverse algorithms can be evaluated and their performance can be compared in the same experimentation environment. Such a study can deliver insights into the feasibility of different strategies.

DES-Chan has been developed as a software framework written in Python. The framework was designed so that it does not require any changes of the Linux kernel or the wireless network interface drivers. Therefore, modifications of the wireless network interface drivers, for instance to reduce the channel switching time, are not part of the framework. DES-Chan is therefore easy to integrate into existing wireless mesh network testbeds.

6.2 Services and architecture

DES-Chan comprises a set of data structures and services useful for channel assignment algorithms. The data structures provided by DES-Chan comprise graph representations of the local network topology and the corresponding conflict graphs [1, 52].

Before we describe the services in detail, we summarize their functionality as follows:

- *Interface handling*: This basic service allows to change the wireless network interface settings, for example to carry out channel switches.
- *Neighborhood discovery*: This service retrieves information about the local network topology and the channel selection therein. This information is usually needed as input for channel assignment decisions in distributed algorithms.
- *Topology monitoring*: A topology monitoring service periodically assesses the local network topology and its state. Neighbors, links, and their respective quality can be monitored which enables the algorithms to become adaptive by reacting to topology changes with refining the channel assignment.
- *Message exchange*: A message exchange service for node to node communication between the instances of an algorithm can be used to disseminate the current channel selection and to negotiate channel switches.
- *Interference modeling*: Interference models are required to estimate the local interference. The models range from simple heuristics such as the $\mathfrak{m}$-hop neighborhood to measurement-based approaches [46, 122].
- *Spectrum sensing*: With spectrum sensing, the conditions of the available channels can be measured. Based on the measurement data, channels may be blacklisted for the channel selection.

The architecture of DES-Chan comprising all modules is depicted in Figure 6.2.1. DES-Chan exposes the offered services and data structures to the channel assignment algorithms. As future algorithms will require additional services, DES-Chan has been designed to be extensible using a modular architecture. As next, the services are described in detail.

6.2.1 Interface management

The interface management module provides wrapper functions to set the configuration of the available wireless network interface cards. The configurations are carried out with *Python WiFi* [129], a library based on the Linux Wireless Extensions. First, the interface can be tuned to a specific channel which is a crucial requirement for all channel assignment algorithms. Additionally, the following settings can be changed: ESSID, cell id, IP address, network mask, data rate, and the transmit power. Further helper functions allow to retrieve the current state of an interface and check for unused interfaces that are still available.

6.2.2 Neighborhood discovery

The *neighborhood discovery* module determines the local network topology by assessing all links and their quality for each network node based on the *expected transmission count*

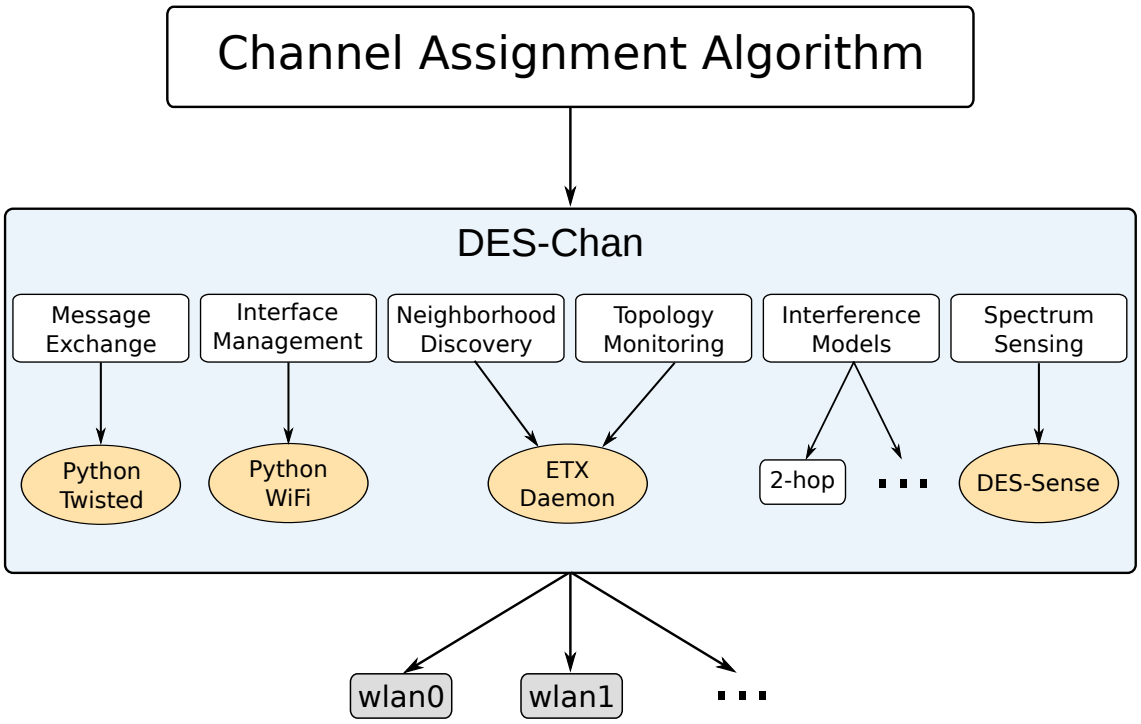


Figure 6.2.1.: Architecture of the DES-Chan framework. DES-Chan provides common services and data structures for distributed channel assignment algorithms.

(ETX) link metric [100]. The ETX link metric estimates how many transmissions for a packet are required until it is successfully received. ETX values are calculated by each node sending broadcast probes and logging how many probes from its neighbors were successfully received. The forward and reverse delivery ratio are then used for the calculation of ETX, because for unicast communication in IEEE 802.11, an ACK frame has to be successfully received at the sender. The ETX value for a link is then calculated as

$$\mathrm{ETX} = \frac{1}{d_f \cdot d_r} \tag{6.1}$$

where d_f is the forward delivery ratio and d_r the reverse delivery ratio.

The ETX implementation `etxd` for the DES-Chan framework has been realized as a Linux daemon. The default settings for `etxd` are summarized in Table 6.2.1. The values can be configured via command line arguments. The probe interval is set to 1 second. The link quality values are averaged over a moving window, which is set to 30 seconds per default, and thus spans 30 probe intervals. The payload of a probe contains all neighbors and the statistics about the received probes from this neighbor according to the window size. The payload length is thus variable, for each neighbor 14 Bytes of data are appended to the probe. The daemon sends UDP probe packets periodically to the broadcast addresses while simultaneously listening for incoming probes of its neighbors. In order to always provide up-to-date information, `etxd` dynamically adapts its configuration if network interfaces have been reconfigured, shut down, or brought up. This is achieved by checking periodically for

Setting	Description
Probe interval	1s
Probe payload	Varying - depends on the number of direct neighbors.
Moving window size	30s

Table 6.2.1.: Default configuration parameters for the `etxd` daemon of DES-Chan.

changes in the interface configurations.

The information of the local network topology can be queried by channel assignment algorithms via an *inter process communication* (IPC) interface that can be accessed via sockets. A simple, textual protocol is used to retrieve the neighbors of a node, as well as the quality and the channel the corresponding link is operated on. The daemon can be queried with an optional parameter regarding a minimum link quality. If used, only those neighbors are returned, which are reached via links for which the ETX value is not higher than the supplied minimum link quality. In addition to the link quality, `etxd` also returns the current channel of the WNIC that is used to reach the respective neighbor. Instances running on other network nodes can also query `etxd` to get all neighbors of a neighbor. Thus, by querying all direct neighbors, each node can retrieve the local 2-hop neighborhood, which is quite commonly used for interference estimation [35].

6.2.3 Topology monitoring

The ETX daemon `etxd` is also the foundation of the topology monitoring service. Via the IPC interface, channel assignment algorithms can query the state of the local network topology periodically and react adaptively to changes. Such topology changes can be caused by the arrival of new nodes or the absence of previously detected nodes. The information about the changes can be used to update the local interference estimation. The local interference level can be calculated repeatedly and may lead to further fine-tuning of the channel assignment decisions.

6.2.4 Message exchange

Node to node communication is an important part of distributed channel assignment algorithms. Messages are exchanged for example to disseminate the current channel assignment of a node. Also, many algorithms require communication protocols to negotiate channel switches in a synchronized manner in order to avoid channel oscillation or dead interfaces [36, 48]. The used communication protocols and message formats can differ significantly depending of the particular channel assignment algorithm. Thus, a flexible networking library is needed, that allows implementing various protocols with few effort.

The Python Twisted [126] library serves as the foundation for the node to node communication. The library provides an asynchronous networking engine and hides technical details like creating sockets and establishing connections from the developer. The researcher can

quickly develop the required protocol implementation for exchanging messages among the network nodes, for instance in order to propagate changes in channel assignment or to carry out three-way handshakes prior to the actual channel switch.

6.2.5 Interference models

The DES-Chan framework supports the implementation of various interference models. Currently, the 2-hop interference model and the *channel occupancy interference model* (COIM), as described in Chapter 5, are available. Based on the 2-hop interference model, for a particular node u, all nodes are regarded as interferers that have a distance of 2-hops or less to u regarding the network topology. The implementation of the 2-hop heuristic supports a binary notion of interference, i.e., two links either interfere or do not. Additionally, a fine-grained notion taking the spectral distance into account is supported [48]. Equation (6.2) shows the cost function I for two center frequencies f_1 and f_2. The additional parameter α denotes the minimum frequency difference of orthogonal channels.

$$I(f_1, f_2, \alpha) = \begin{cases} 0 & \text{if } |f_1 - f_2| \geq \alpha \\ 1 - \frac{(|f_1 - f_2|)}{\alpha} & \text{otherwise} \end{cases} \tag{6.2}$$

By setting α to 30 MHz in the interference cost function I, three channels of the IEEE 802.11b/g 2.4 GHz frequency band are theoretically orthogonal, i.e., $\{1, 7, 13\}$.

For COIM, the results of the measurements are stored on each network node. Each row comprises the measured channel occupancy at a testbed node while another node was sending at full capacity. The DES-Chan module for COIM accesses the database on demand to retrieve the corresponding values for the channel occupancy measurements. Based on the values, the channel assignment algorithm instance on one particular node can determine if another node is an interferer and if so, the channel selection of that node should be considered for its own channel assignment decisions.

6.2.6 Spectrum sensing

Spectrum sensing is an efficient method to measure the load on the available channels and thus detect the radio activity of external networks and devices. Especially with the rising number of wireless devices in the unlicensed frequency spectrum, solutions to counter the *spectrum scarcity* problem is becoming more important [130]. DES-Chan provides the DES-Sense module to carry out measurements of the channel load on specified channels. While the spectrum sensing method is describe in detail in Section 8.2, we describe here how the channel assignment algorithms can measure and retrieve the channel load values.

The DES-Sense module is a software-based spectrum sensing solution and consists of the following two components:

- *Sensing component* - The sensing component is a daemon that periodically measures the channel load based on the carrier sensing statistics of the wireless network card.

Based on the statistics we are able to calculate the channel busy time, in which the wireless medium could not have been used for wireless transmissions. The sensing component can be configured dynamically with the set of channels $C = \{c_1, c_2, .., c_k\}$ that will be monitored and the duration $T = \{t_1, t_2, .., t_k\}$, each channel is monitored.

- *IPC interface* - For the integration into DES-Chan, an *inter process communication* (IPC) interface is provided that allows algorithms to retrieve the measured statistics about the channel load. The results of the most current measurements can be queried via the IPC interface so that the channel assignment algorithms can fuse the statistics into their channel assignment decision.

The particular channel assignment algorithm, can decide if the channel load is only measured prior to the execution of the algorithm or also during its operation. Since the network nodes are not equipped with a dedicated interface for the spectrum sensing measurements, one interface per node can not be used for other traffic during the measurement phase.

6.2.7 Data structures

DES-Chan provides data structures and functions for graph representation to model the network graph. A data structure for the network graph contains a vertex for every network node and an edge for every link between two network nodes. The data structure allows to store a list of channels for each edge, thereby supporting multiple links between node pairs. Functions are provided to access the vertices and edges, for instance to update the channel selection. Graph objects can be stored in the DOT format, that can processed to generate an image of the graph [131]. A function provides the possibility to merge two graph objects together by creating the union of both vertex sets and edge sets. This is useful, if the 2-hop neighborhood of a particular node is constructed by querying the direct neighbors for their local network topology information.

Additionally, a corresponding data structure for conflict graphs is provided. The conflict graph is constructed from the network graph and the specified interference model. Each edge of the network graph is a vertex in the resulting conflict graph. An edge between two vertices of the conflict graph exists, if the corresponding links interfere with each other according to the applied interference model. The data structure for conflict graphs provides the corresponding transformation methods for these actions. It allows to manipulate the channel of the corresponding edge in the network graph and automatically updates the interference information in the conflict graph. The conflict graph data structure can be easily extended to implement other concepts, such as multi-radio conflict graphs [53].

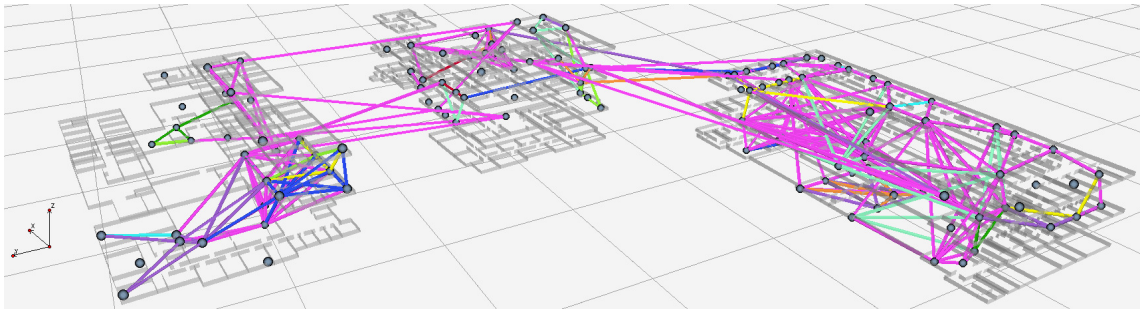


Figure 6.2.2.: Snapshot of the visualization tool DES-Vis. The screenshot displays the network topology of the DES-Testbed. Links are displayed in different colors depending on the channel they are operated on.

6.3 Channel assignment visualization

A visualization of the channel assignment procedure and the corresponding network topology can be helpful in analyzing the performance of channel assignment algorithms. In order to visualize the channel assignment, we extended DES-Vis, the 3D-visualization tool of the DES-Testbed Management system (DES-TBMS) [96]. DES-Vis visualizes the state of the network based on the DES-Mon network monitoring tool. For each experiment, a monitoring interval can be set, in which DES-Mon gathers snapshot-based the state of each network node. The data that is retrieved by DES-Mon has been extended with the information of the local network topology as assessed by the ETX daemon `etxd`. This data comprises for each node the links to all direct neighbors, including the channel the link is operated on. Thus, for each snapshot, the global channel assignment is retrieved.

Based on this information, the DES-Vis was extended to display links in different colors depending on the utilized channel. This way, a graphical representations of the channel assignment can be displayed and a first insight of the achieved channel diversity is possible. An example of the visualization of the network state is shown in Figure 6.2.2.

6.4 Extensibility

The modular architecture of DES-Chan allows an easy integration of additional services. While core services such as the interface management and the message exchange module will most likely stay untouched, alternatives for other modules may offer interesting insights. Especially the availability of different interference models allows to compare the performance in terms of the accuracy of the interference estimation. Currently, DES-Chan comprises the 2-hop interference model and the measurement-based *channel occupancy interference model* (COIM). Evaluating channel assignment algorithms with different algorithms may deliver insights on the potential of more accurate interference models. Such an evaluation is presented later in Section 7.5.

CHAPTER 7

A study of link-based and interface-based channel assignment

In this chapter, we present a study of the performance of existing distributed channel assignment algorithms in a large-scale network environment. As first, we describe two distributed algorithms in detail with the focus on the developed implementation candidates based on DES-Chan. A performance evaluation of these algorithms in a large-scale wireless network follows. As part of the study, the performance of the algorithms is also compared to a randomized approach and the single channel network. The chapter closes with a discussion of the results and an outlook towards the external-interference aware channel assignment. Excerpts of this chapter have been published in [119].

7.1 Motivation

The motivation of this experimental study is to investigate the performance gain with distributed channel assignment in large-scale networks. We compare the performance of two promising approaches from literature that have achieved good results in smaller testbeds with less than 15 network nodes. Both approaches are distributed algorithms that optimize the channel assignment in the local neighborhood of each network node in a simple greedy manner. However, there is a fundamental difference in how channels are assigned.

The *minimum interference channel assignment* (MICA) algorithm [36] assigns channels to links. This *link-based* strategy has the benefit that the channel assignment is independent of the overlaying routing algorithm because all links in the single channel network also exist in the multi channel network. The *distributed greedy algorithm* (DGA) is an *interface-based* approach [48, 62], in which channels are assigned to interfaces. The connectivity of the network is preserved by one dedicated interface per node operating on a common global channel as a fall back if no other link on a different channel is available. The interface-based approach promises to be more robust in dynamic topologies than the link-based approach, albeit the dedicated interface can not be used for channel assignment.

Both algorithms have been evaluated by their authors in small wireless testbeds (with 11 and 14 multi-radio nodes) and produced significant improvements in regard to the network capacity. The experimental study in this chapter is guided by the following three questions:

- Do the reported performance results in small wireless testbeds also hold in a large-scale wireless testbed with more than 100 nodes?
- Based on the performance evaluation with the channel assignment benchmark, is one of the two strategies more favorable in large-scale wireless networks?
- Are such simple greedy strategies suitable as a foundation for our envisioned external-interference aware channel assignment?

We first describe both algorithms and their implementation candidates for the DES-Testbed based on DES-Chan followed by the performance evaluation of the algorithms based on the channel assignment benchmark (see Chapter 4).

7.2 Minimum Interference Channel Assignment (MICA)

7.2.1 Idea

The *minimum interference channel assignment* (MICA) assigns channels to links instead of interfaces and is therefore topology preserving [36]. In other words, all links in the single channel network, also exist in the multi channel network after the channel assignment procedure. This way, the approach is independent of the overlaying routing algorithm. The authors formulate the channel assignment problem as minimizing the *total network interference* by minimizing the number of edges in a conflict graph [1,52]. The problem is reduced to the Max K-cut problem [63], thus proving that it is NP-hard.

7.2.2 Algorithm

The proposed distributed algorithm is based on a greedy approximation algorithm for the Max K-cut problem given in [63]. The distributed MICA algorithm has been presented in detail in [36]. At the network initialization, all links are assigned to the same channel. After retrieving the local communication graph of the m-hop topology (with $m \geq 1$) and constructing the conflict graph V_c, each nodes tries to minimize the local interference. Each node iterates over all link-channel combinations (u, c), identifying the combination which results in the largest decrease of interference in the local neighborhood. The largest decrease is achieved with the combination (u, c) that removes most edges in the local conflict graph V_c compared to all other combinations.

The interface constraint has to be respected, which means that no more channels can be assigned to a node than it has interfaces. In order to avoid channel oscillations, additional constraints for the channel switches have been introduced. A node may only switch the channel of a link that is incident to that particular node. Additionally, a network link

between two nodes is owned by the node with the higher ID and only this node may assign a channel to the link. The node ID can be any unique identifier, such as IP address or MAC address. In summary, in each iteration at node n, the link-channel combination (u, c) is selected, that fulfills all of the following criteria

- Node n is incident to the link u
- Node n *owns* link u, meaning that its ID is higher than the ID of the other node incident to u
- Applying channel c does not violate the interface constraint on both nodes
- The combination (u, c) has not been tried before at node n
- The combination (u, c) results in the highest decrease of interference in regard to the number of edges removed in the conflict graph V_c

If no further (u, c) combination can be found that matches all criteria, the algorithm terminates. Once a combination (u, c) is found, node n sends a `ChannelRequest` message to the corresponding neighbor m for the link u. If changing the channel on u to c does not violate the interface constraint, node m replies with a `ChannelReply` message and switches the channel accordingly. Otherwise node m does not reply to the `ChannelRequest` message. After receiving the `ChannelReply` message, node n changes the channel as well and sends an `ChannelUpdate` message to all of its m-hop neighbors, so they can update their local conflict graphs. If the `ChannelReply` message has not been received at node n until a timeout, node n discards the combination (u, c) and proceeds with the next iteration.

The pseudo code for the distributed MICA algorithm running on one particular node is given in Algorithm 2. Additionally, asynchronous events may occur, for instance when a node receives a *ChannelRequest* message from a neighbor. These cases are described in the next section with a flow diagram of the implementation candidate in Figure 7.2.1.

The termination of the algorithm is ensured with the constraint that each link-channel combination (u, c) can only be tried once. This also gives insights into the runtime of the algorithm. The total number of iterations (over all instances of the algorithm) is $O(|V_c| * K)$, where $|V_c|$ is the number of vertices in the conflict graph (of the whole network) and K is the number of available channels.

7.2.3 Implementation

A first implementation based on DES-Chan has been created in the scope of a master thesis in [132]. The implementation candidate comprises two main classes for the algorithm and the communication between individual instances: *MICA* and *Messaging*. Since the approach is a distributed algorithm, it is executed concurrently on every node. The following description refers to an instance running at one specific node. Before the algorithm is started, the single channel network is initialized by setting one of the network interfaces to

Algorithm 2 Distributed MICA running on node n

Require: local network graph V, local conflict graph V_c

```
1: S ← all possible link-channel combinations (u, c)
2: repeat
3:     minimum ← 0
4:     for (u, c) in S do
5:         if n not owner(u) then
6:             continue
7:         end if
8:         if switching u to c violates the interface constraint then
9:             continue
10:        end if
11:        if f_I(u, c) > f_I(minimum) then
12:            // interference reduction with (u, c) is larger than current best choice
13:            minimum ← (u, c)
14:        end if
15:    end for
16:    if minimum > 0 then
17:        send ChannelRequest to corresponding neighbor of u
18:        if ChannelReply received then
19:            switch u to c, update V_c
20:            send ChannelUpdate to m-hop neighborhood
21:        end if
22:        S ← S \ (u, c)
23:    end if
24: until minimum = 0
```

the common global channel. An instance of the MICA class is created and initialized with an empty conflict graph and a new Messaging object. From that point on, the program flow is determined by a series of events, which are depicted in the flow diagram in Figure 7.2.1.

The Messaging object listens for incoming messages from other MICA instances, and uses the *Neighborhood Discovery* module from DES-Chan to retrieve the initial network graph. The local conflict graph, which is still empty at that point, is updated with the information of the network graph. Afterwards, the *findMinimum* method is called. The method determines the link-channel combination (u, c) that leads to the largest decrease in interference. If such a link-channel combination is found, a `ChannelRequest` message is sent to the corresponding neighbor of link u. If no combination could be found that further reduces the interference, the algorithm terminates.

After the `ChannelRequest` message has been sent, the program flow is stalled until a message from the corresponding MICA instance is received or a timeout is reached. The reply can either be a `ChannelReply` or a `ChannelReject` message. We introduce the `ChannelReject` message, which is not part of the original algorithm, in order to avoid long waiting times if the combination (u, c) violates the interface constraint at the receiving node. If the `ChannelRequest` is answered with a `ChannelReply`, the originator of the request applies the new channel to the particular link. To make sure that the channel

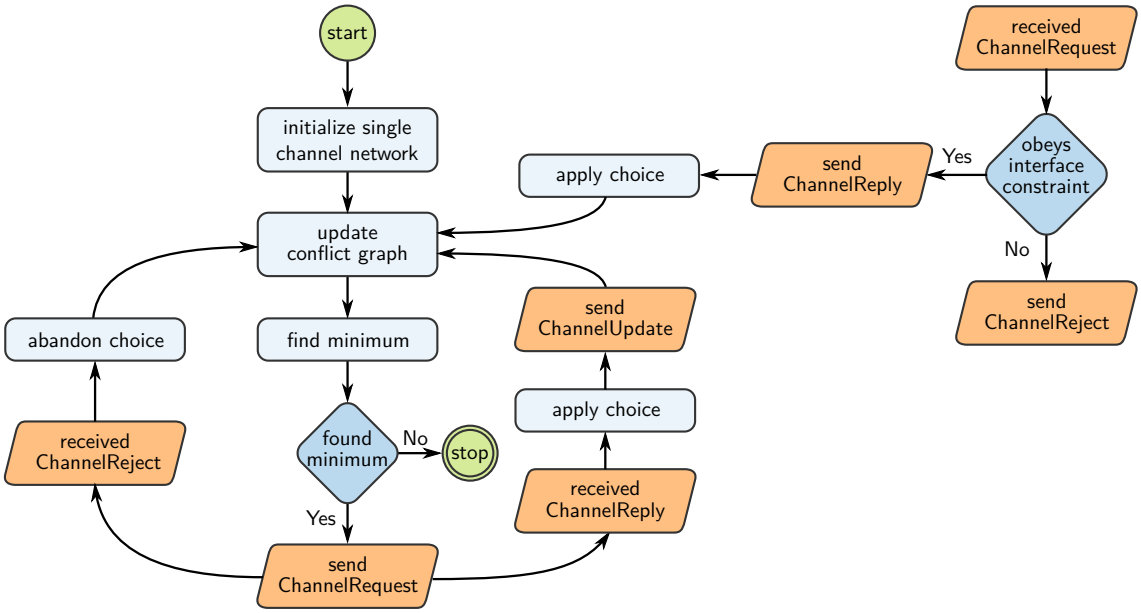


Figure 7.2.1.: Flowchart of the distributed MICA implementation. Methods of the *MICA* class are colored light-blue, receiving and sending messages are represented by light-orange parallelograms.

can actually be switched to when the `ChannelReply` message is received, the program had allocated a lock for the unused interface before sending the `ChannelRequest`. Thereby it ensures, that the interface is not used to satisfy a `ChannelRequest` from another neighbor, while waiting for the corresponding `ChannelReply` message as described below. After receiving the `ChannelReply`, the node applies the pending channel switch and sends a `ChannelUpdate` message to its m-hop neighbors which update their local topology information. If the channel request is answered with a `ChannelReject` message, the respective link-channel combination is removed from the set of possible combinations. It will not be considered again by the *findMinimum* method, which ensures the termination of the algorithm. After the requested channel has been either approved or rejected, the current network topology and the local conflict graph are updated accordingly. Afterwards, the next iteration is started until the interference cannot be decreased any further.

During the execution of the algorithm, `ChannelRequest` messages from other instances may be received asynchronously. When receiving such a `ChannelRequest`, the node checks whether it is able to change the channel without violating the interface constraint. If the corresponding network interface is only used for communicating with the sender of the request, the interface constraint is respected and the link can be switched to the new channel. If, however, other neighbors are connected via the same network interface, the proposed channel switch would eliminate these links. In that case, an unused network interface has to be set up and tuned to the respective channel. If there is an unused interface available, or no other links would be affected by the channel switch, the node sends a `ChannelReply` message, indicating that the channel can be changed. After the reply has been sent, the node configures its interfaces according to the new assignment.

In contrast, if the channel switch would affect links to other neighbors and no unused interfaces are left, the request is answered with a `ChannelReject` message and the link-channel combination is discarded.

7.2.4 Deviations from the original algorithm

The implementation varies slightly from the original algorithm. Explicit `ChannelReject` messages have been introduced in the presented implementation. When a node receives a channel request, which cannot be fulfilled because of the interface constraint, it replies with a `ChannelReject` instead of not replying at all. Thereby, long waiting times are avoided and the node can immediately continue with the next iteration.

7.2.5 Feasibility of link-based channel assignment

Fractional Network Interference (FNI)

MICA has been evaluated in a 11 node testbed by the original authors. The evaluation included a graph-theoretical analysis based on the *fractional network interference* (FNI). The FNI is defined as the ratio of the number of edges of the conflict graph of the single channel network and the number of edges in the conflict graph after the final channel assignment. More formally, the FNI is defined as

$$\mathrm{FNI} = \frac{|\mathrm{E}_{\mathrm{CG}_{\mathrm{CA}}}|}{|\mathrm{E}_{\mathrm{CG}_{\mathrm{Single}}}|} \tag{7.1}$$

with $|\mathrm{E}_{\mathrm{CG}_{\mathrm{CA}}}|$ as the number of edges in the conflict graph after the channel assignment, and $|\mathrm{E}_{\mathrm{CG}_{\mathrm{Single}}}|$ as the number of edges in the single channel network. The authors reported for a network with 3 wireless network interfaces per node a FNI of about 0.35.

In our experimental evaluation (see Section 7.5), the FNI for MICA over 35 experiment replications is shown in Figure 7.2.2. The median over all experiment replications for the FNI is about 0.5, meaning that the amount of interference relationships has been halved due to the channel assignment procedure. However, the value is also significantly higher than what the authors reported for a similar setup. The reasons for this deviation are the following. First, the authors calculate the graph-theoretic FNI from generated static topologies without using a network simulator or testbed for the evaluation. In our experiments, the network topology is not static and the quality of links varies over time, even if we only consider high quality links for the channel assignment procedure. For this reason, links may break or new links may emerge during the channel assignment procedure. From the log files of the execution of MICA, nodes have been removed from the set of interferers (and neighbors) or added at a later point of time. If a MICA instance does not have complete and accurate knowledge of the neighbors and interferers of a node, the results are likely to be not optimal.

Second, the network density of the DES-Testbed has been increased with the latest testbed extension. Additional DES-Nodes have been placed in the already covered build-

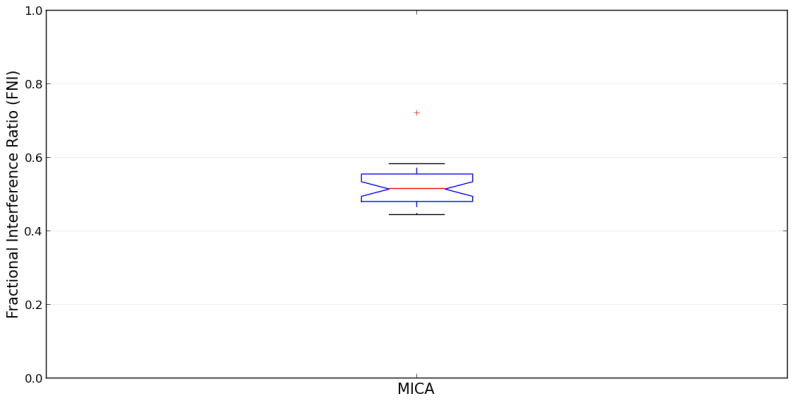


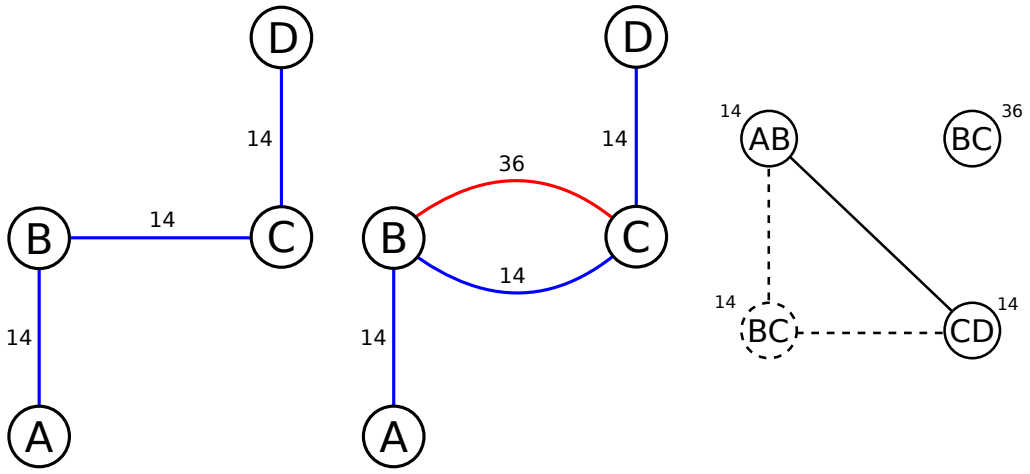
Figure 7.2.2.: Fractional Interference Ratio (FNI) for MICA. The median of the FNI over all experiment replications is about 0.5.

ings, thus there are several nodes with a relative high node degree higher than 15. With the increased network density, the number of edges in the conflict for the single channel network increases as well. Still, the number of wireless interfaces for the DES-Nodes has not changed. For this reason, the majority of edges in the conflict graph adjacent to a highly connected node can not be removed. These reasons can explain, while the values for the FNI in our experimental study are higher than the ones originally presented in the paper.

Observed Issues

The analysis of the first results with the MICA algorithm in the DES-Testbed revealed some interesting observations. The first issue arises from the fact that the original algorithm considers only one link between two nodes in the network. Nodes are either connected or not, but multiple links between the same nodes using different radios are not taken into account. The algorithm is topology preserving, thus links should neither break or emerge during the execution of the algorithm. However, during the channel switch procedure, an *implicit link* may be created, that is not considered by MICA. For example in Figure 7.2.3, after retrieving the initial network topology, the algorithm changes the link (BC) to channel 36. However, both nodes are connected via the interface tuned to channel 14, which is still used to communicate with their neighbors A and D respectively. Therefore, an implicit link on channel 14 exists between B and C. In the view of the algorithm the edges in the conflict graph in Figure 7.2.3c shown as dotted lines have been removed with the channel switch. But since the resulting network graph is a multi-graph, these edges in the conflict graph still exist and an interference-aware routing has to be used in order to not utilize the implicit link. These implicit links violate the feature of the algorithm that it is topology preserving in a way that it is transparent for routing algorithms.

Another issue arises from the assumption of link stability in real wireless networks. For the channel assignment procedure, the authors consider only stable links with a *packet deliver ratio* (PDR) of at least 80%. However, we observed that some wireless links are



(a) Initial network topology (b) Channel switch for link BC (c) Resulting conflict graph

Figure 7.2.3.: Implicit links with MICA. Starting from the initial network graph using channel 14 in (a), the link BC has changed to channel 36 in (b). In order to preserve the links AB and CD, a previously unused interface has been used to apply the new channel assignment, which leads to an *implicit link* on channel 14 between the nodes B and C. In the view of the algorithm the edges in the conflict graph in (c) shown as dotted lines have been removed with the channel switch.

very dynamic even in stationary wireless mesh networks such as the DES-Testbed [91]. Their quality may change over time and they exhibit different characteristics on different channels. Thus, existing links may break apart and new links may emerge during the course of the algorithm. Since the algorithm does not account for that, the resulting channel assignment may be sub-optimal. In the worst case, channels are assigned to links which temporarily do not exist and thus can not be used. As a solution, the current network topology could be updated periodically to notice recent changes.

Considering only high quality links also has the consequence that poorly connected nodes may not be reached. With a growing network size, the probability rises that nodes exist on the network edge that are connected via lossy links. Selecting only high quality links for the channel assignment procedure may thus exclude a fraction of nodes. The already lossy links may break if nodes change their channels, resulting in network partitioning, which is clearly not desired. As a solution, the constraint for the link quality can be relaxed to also take lossy links at the network edge into account.

7.3 Distributed Greedy Algorithm (DGA)

7.3.1 Idea

The *distributed greedy algorithm* (DGA) assigns channels to network nodes, or more precisely, to network interfaces [48,62]. Intuitively, each node tries to minimize the interference

by assigning the least used channels in its *interference set*. The interference set of a node n consists of all nodes and their channel assignment whose transmissions affect sending and receiving at node n. To preserve the network connectivity, one interface on every node is operating on a global common channel and not used for channel assignment. For any additional interfaces (referred to as *switchable* interfaces), the channel is determined with the following algorithm.

7.3.2 Algorithm

At the network initialization, one dedicated interface is set to the common global channel. After retrieving the local communication graph, the interference set of each node is determined, which is approximated with the m-hop neighborhood (with $m = 3$ in the original publication[1]). With an *interference cost function* f_I as given in Equation (7.2), the spectral overlap of two channels is used to determine the level of interference in the interference set. The cost function takes the center frequencies f_1 and f_2 of two radio channels and the additional parameter α into account. α denotes the minimum frequency difference of orthogonal channels. A value of $f_I = 0$ means that the channels are orthogonal, meaning they do not interfere with each other.

$$f_I(f_1, f_2, \alpha) = max(0, \alpha - |f_1 - f_2|) \tag{7.2}$$

Using the interference cost function, each node tries to assign the channel that results in the lowest level of interference in its interference set to its switchable interfaces according to Algorithm 3. In each iteration, the algorithm considers exactly one switchable interface and calculates the interference cost for all available channels K with the channel assignment of the interference set. The results is the channel c_{best} with the lowest overall interference cost. If c_{best} is different than the currently assigned channel, a channel switching procedure is initiated. As an additional constraint, only channels can be assigned that are used by at least one neighboring node, in order to avoid isolated interfaces.

The channel switching procedure is based on a 3-way handshake with the nodes in the interference set to synchronize channel switches [133]. When a node wishes to change a channel, it sends a `ChannelRequest` message to all nodes in its interference set. The request contains the old channel for the interface in question, the new channel the node intends to assign, the achieved reduction of interference, and the channel assignment of the receiving node. The latter is used to verify, that the channel reduction has been calculated based on the correct assumptions of the channel assignment in the interference set. When receiving a `ChannelRequest` message, a node puts a possible own channel switch intent on hold until the ongoing procedure is finished. The recipients of the `ChannelRequest` reply with a `ChannelAccept` message, if the proposed channel switch does not conflict with their own intended channel switches. The originator of the request applies the channel switch,

[1] The results of the experiments with the *channel occupancy interference model* in Chapter 5 showed that considering the 3-hop neighborhood is too pessimistic for the DES-Testbed.

Algorithm 3 Channel selection with DGA for an interface n_i at node n

Require: $S_n \leftarrow$ Interference set of node n
$K \leftarrow$ Set of available channels
$c_j \leftarrow$ channels assigned to each node $j \in S_n$
$c_{n_i} \leftarrow$ current channel of the interface n_i
1: $F_{best} \leftarrow \sum_{j \in S_n} f_I(\alpha, c_{n_i}, c_j)$
2: $c_{best} \leftarrow c_{n_i}$
3: **for** k **in** K **do**
4: $F(k) \leftarrow \sum_{j \in S_n} f_I(\alpha, k, c_j)$
5: **if** $F(k) < F_{best}$ **then**
6: $F_{best} \leftarrow F(k)$
7: $c_{best} \leftarrow k$
8: **end if**
9: **end for**

if all nodes of the interference set reply with a `ChannelAccept` message. A following `ChannelUpdate` message is sent to the nodes in the interference set to confirm the channel switch. If the `ChannelRequest` conflicts with another pending channel switch, a node replies with a `ChannelReject` message. When receiving a `ChannelReject` message, the originator of the request aborts the current channel switching procedure by sending a `ChannelAbort` message to all nodes in its interference set.

The algorithm terminates, when the local interference can not be reduced any further. The authors prove the convergence of the algorithm by showing that the overall network interference decreases monotonically with each channel switch [62]. It has to be stated though, that this does only hold in static network topologies.

7.3.3 Implementation

A first implementation based on DES-Chan has been created in the scope of a bachelor thesis in [134]. Similar to MICA, the implementation candidate of DGA is distributed into two main classes. The *DGA* class comprises the channel selection algorithm while the *Messaging* class contains the implementation of the communication protocol for channel switch synchronization. When the algorithm is started, one interface on each network node is set the common global channel to create a single channel network. The switchable interfaces are initialized by selecting a channel randomly from the set of available channels.

As next, information about the local 2-hop neighborhood of each node is retrieved with the *Neighborhood Discovery* service of the DES-Chan framework using the ETX metric [100]. Once the local topology has been retrieved, the interference set is created by querying all 2-hop neighbors for information about their current channel assignment. This completes the initialization after which the program flow is determined by a series of events, which are depicted in the flow diagram in Figure 7.3.1.

First, the channel selection algorithm for one of the switchable interfaces is executed. If a better channel than the current one is found, the node generates a `ChannelRequest` and

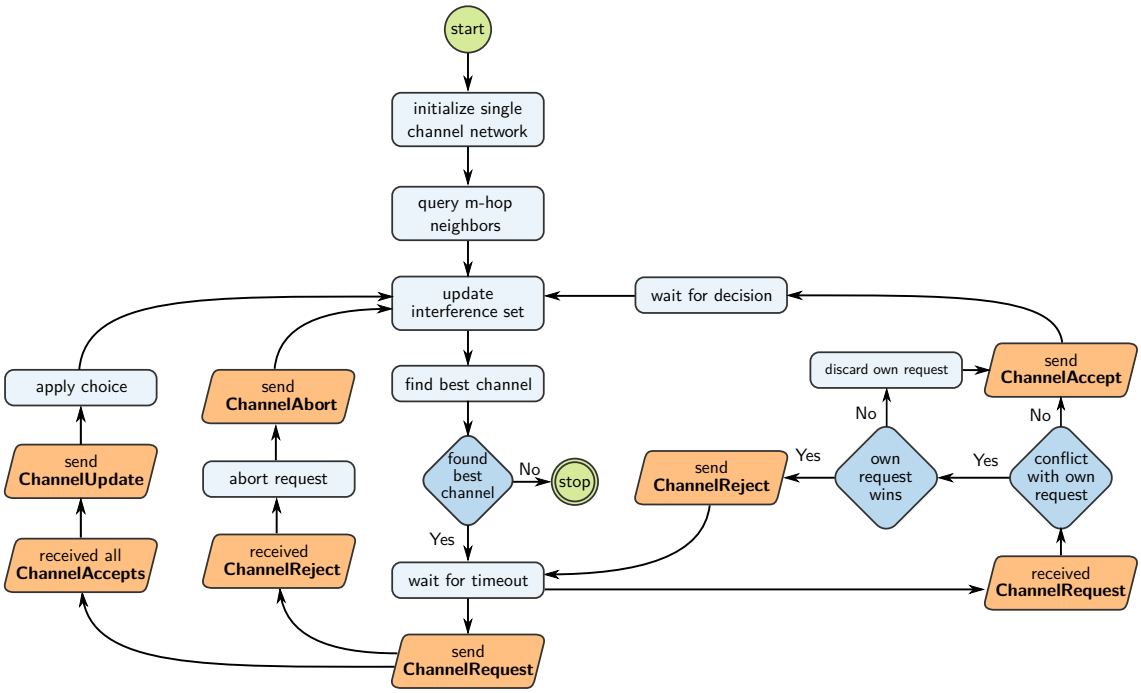


Figure 7.3.1.: Flowchart of the distributed DGA implementation. Methods of the *DGA* class are colored light-blue, receiving and sending messages are represented by light-orange parallelograms.

calculates an exponentially distributed delay before initiating the channel switch procedure. During the waiting time, `ChannelRequests` of other nodes may be received and answered accordingly with a `ChannelAccept` or `ChannelReject`. A `ChannelReject` is sent, if the received request either has false assumptions of the node's current channel assignment or the request conflicts with another one and has a lower priority. The priority of a request is determined by the reduction of interference it achieves according to the interference cost function. If two competing request achieve exactly the same level of interference reduction, the request originated by the node with the higher ID has the higher priority. Otherwise, the incoming request is replied with a `ChannelAccept` and the node awaits the final decision of the channel switch procedure. The channel switch procedure is successful, if the originator of the request receives `ChannelAccept` messages from all nodes in its interference set. Finally, a `ChannelUpdate` message is sent to the nodes in the interference set and the channel switch is applied. If a `ChannelReject` message has been received or a timeout is reached, the channel switch procedure is aborted. A `ChannelAbort` message is sent and the next iteration of the algorithm begins.

7.3.4 Deviations from the original algorithm

The original specification of the algorithm in [48] considered two network interfaces per node resulting in exactly one switchable interface per network node. While the authors do not restrict their approach to network nodes equipped with exactly two radios, several

unclear issues arise when nodes with additional radios are considered.

- *Same channel restriction:* First, it is obviously not desired that multiple switchable interfaces are tuned to the same channel. To prevent that a node assigns the same channel to more than one interface, we introduce an additional constraint.

- *Initial channel selection:* The channel selection algorithms limits the available channels to only those, which are used by at least one of the direct neighbors of a node. An important question is the initial channel selection for the switchable interfaces. Obviously, they can not be initialized using the common channel, because no channel diversity would be possible. We solved this issue, by randomly assigning channels to the switchable interfaces before the first call of the channel selection algorithm.

- *Interface priority:* In each iteration of the channel selection algorithm, only one interface is considered. The order of the interfaces in which they are considered for the channel assignment calculation is not specified. The priority of the interfaces can either be solved in a round robin manner, or greedily considering additional parameters. For instance, the algorithm can prioritize the interface that is tuned to the most used channel in the interference set, for which a channel switch would most likely lead to the largest decrease in interference.

- *Interference cost functions for multiple bands:* In the original algorithm description, the common global channel was chosen from one frequency band (2.4 or 5 GHz) and the available channels for the switchable interface were limited to the channels of the other band. With multiple switchable interface, selecting multiple channels of one frequency band can not be avoided. The experiments investigating the *adjacent channel interference* (ACI) revealed that different values for α at 2.4 and 5 GHz are valid in the DES-Testbed (30 and 60 MHz, respectively). These values have to be considered when the interference level is determined with the interference cost function.

The implementation candidate has been adapted with the required changes.

7.3.5 Feasibility of interface-based assignment

An interesting aspect of the algorithm is the relatively high protocol overhead the algorithm imposes on the network. In contrast to MICA, where only the two nodes incident to the link in question negotiate a channel switch, all nodes of the interference set participate in the channel switch negotiations. The 3-way handshake with all nodes of the interference set is required to guarantee the convergence of the algorithm. Verifying the channel assignment of the interference set before applying channel switches ensures that the decisions are based on the correct assignment. This way, semi-optimal decisions based on incomplete or obsolete information are avoided.

7.4 Random channel assignment (RAND)

The *random channel assignment* (RAND) algorithm realizes a simple random strategy that functions as follows. One interface per node is set to a common global channel in order to preserve the network connectivity. Any additional interface on a network node is set to a random channel of all available channels in the 2.4 and 5 GHz frequency spectrum. However, we introduce the additional constraint that any two interfaces on a node do not use the same channel.

A randomized channel assignment algorithm has been often used in order to compare its performance to other more sophisticated approaches. However, depending on the network topology and the number of available channels, such a randomized approach may already be capable to increase the network capacity significantly. As an example for such a strategy, the approach in [69] uses a randomized component when it is guaranteed that two nodes operate on a mutual channel.

7.5 Performance evaluation

The performance evaluation has been carried out with the *channel assignment benchmark* (CAB) described in Chapter 4. The DES-Testbed functioned as experimental platform.

7.5.1 Experiment settings and runtime

The following algorithms have been evaluated in this experimental study: MICA, DGA, RAND, and the single channel network (SINGLE) as a baseline. The settings and parameters for all experiments are given in Table 7.5.1. One experiment replication consisted of measuring the performance metrics of the channel assignment benchmark IAR, IRR, PO, and STR for all algorithms in question. Additionally, we discuss the resulting network topology after the channel assignment and the channel distribution of the wireless links. First, we compare the algorithms utilizing the 2-hop model for interference estimation. In a second step, we compare the performance of DGA utilizing the 2-hop model and the *channel occupancy interference model* (COIM). The runtime for one experiment replication with all scenarios of CAB was about 160 minutes. With a total number of 35 replications, the whole study took about 93 hours.

7.5.2 Network topology and link distribution

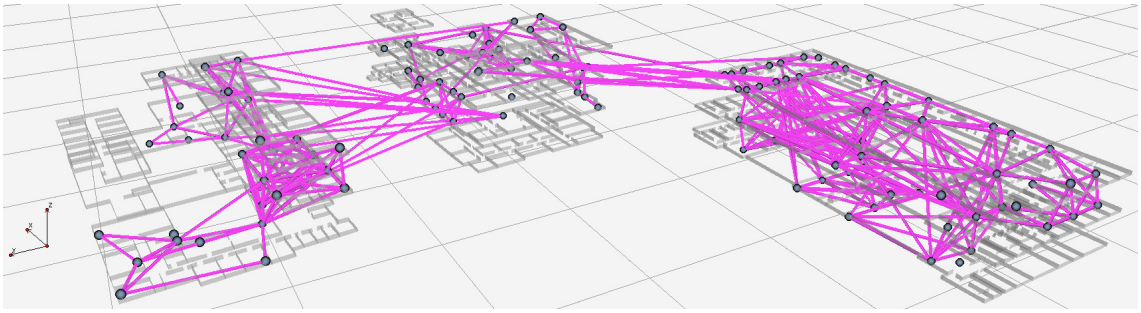
An snapshot of the network topology after the channel assignment procedure for each algorithm is depicted in Figure 7.5.1. For the single channel network (SINGLE) in Figure 7.5.1a, all links are operated on the default channel. In the resulting network topologies for all other algorithms, links are also operated on other channels while the default channel is still predominant. For DGA and RAND the reason is that one interface per node is permanently tuned to the default channel in order to preserve the network topology. MICA

Parameter	Values	Comment
Number of nodes	126 DES-Nodes	Each node is equipped with one IEEE 802.11b/g radio and two IEEE 802.11a/b/g radios.
Experiment replications	35	Each replication ran in 160 minutes, resulting in a total experiment duration of about 93 hours.
Algorithms	MICA, DGA, RAND, SINGLE	The single channel network (SINGLE) functions as a baseline.
Algorithm runtime	20 minutes	Most instances completed the assignment in that time.
Available channels	36, 44, 48, 52, 60, 64, 100, 108, 112	This applies to MICA and DGA. The randomized approach uses all channels of the 2.4 and 5 GHz bands.
Default channel	14	Applies to MICA, DGA, and RAND. The single channel network is initialized on this channel as well.
Interference model	2-hop model, COIM	COIM has been developed in the scope of this dissertation and is described in Chapter 5.
Traffic generator	`nuttcp`	See Section 4.3.8.
Performance metrics	IAR, IRR, SAT, PO	Additionally, we consider the final network topology and channel usage distribution.

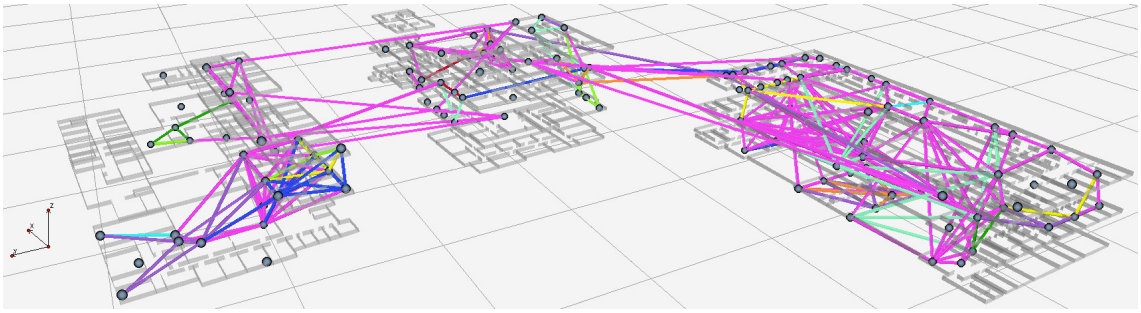
Table 7.5.1.: Experiment settings for the comparison of channel assignment algorithms.

starts with setting up one interface per node on the default channel as well, but allows to change this channel for a link, if a pending channel switch would not cause losing the direct connection to a neighboring node. This means in practice, that only nodes with a node degree of 3 or less are able to switch the default channel to another one because otherwise links will break. This is because of the *interface constraint*, which states that each node can only assign as many channels as it has network interfaces. Since the average node degree is about 6 for all experiment replications, most nodes operate the initial interface permanently on the default channel as well.

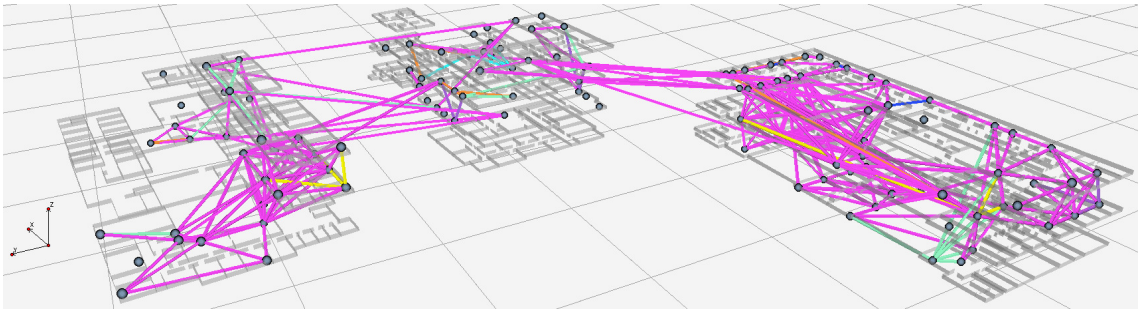
As next, we investigate the spectral diversity created by the algorithms considering the number of links of the resulting network topology. The number of unidirectional links per algorithm in the network graph with a quality of $\frac{1}{\mathsf{ETX}} \geq 0.8$ is depicted in Figure 7.5.2a. The plot shows, that all channel assignment algorithms were able to increase the number of links compared to the single channel network. In the single channel network, 600 links have been established in the median of 35 experiment replications. For RAND, 710 links have been determined, 750 links for DGA, and 780 for MICA. This is as expected, since all algorithms preserve the network topology, and use the multiple WNICs to create additional



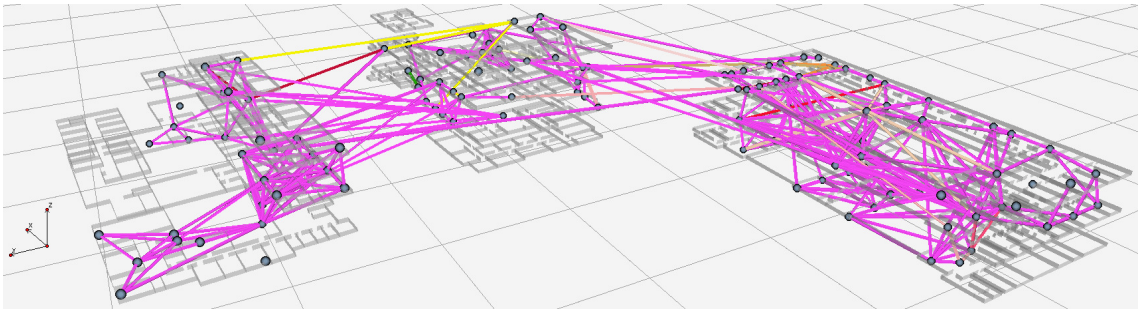
(a) The single channel network (SINGLE)



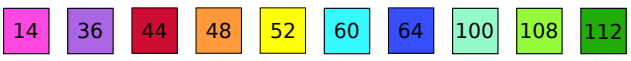
(b) Final channel assignment with MICA



(c) Final channel assignment with DGA

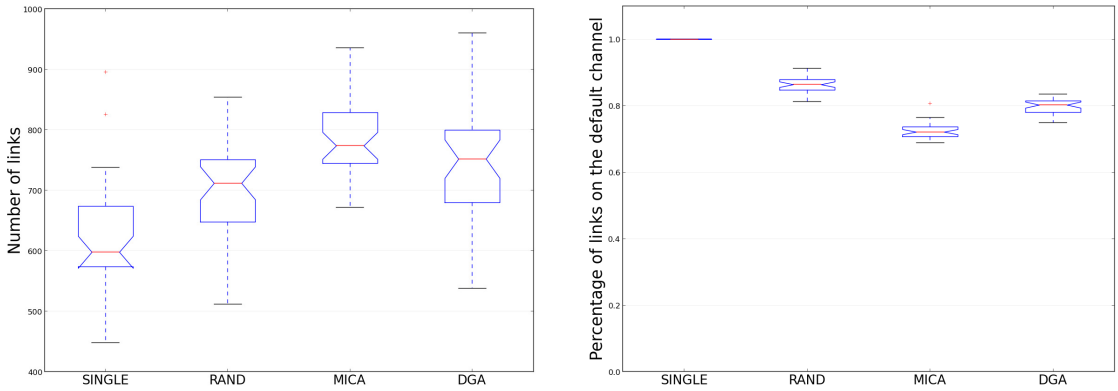


(d) Final channel assignment with RAND



(e) Channel color legend

Figure 7.5.1.: Snapshot of the network topology of the single channel network (SINGLE) after the channel assignment for MICA, DGA, and RAND.



(a) Number of uni-directional links per algorithm with $\frac{1}{\text{ETX}} \geq 0.8$.

(b) Percentage of links operating on the default channel.

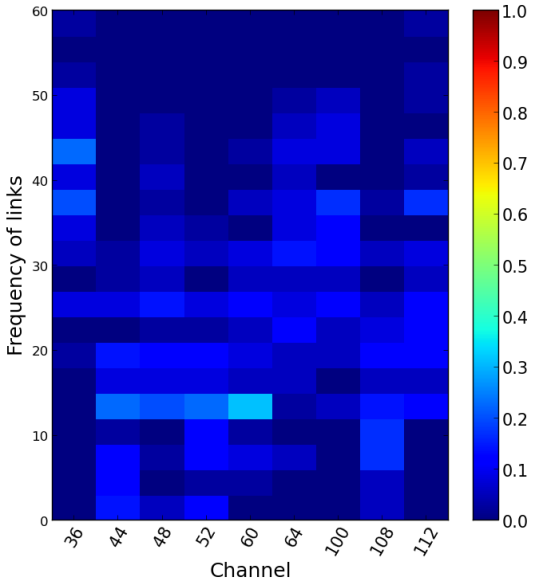
Figure 7.5.2.: Amount of unidirectional links in the network and the percentage of links operating on the default channel. Figure 7.5.2a shows that compared to the single channel network (SINGLE), the number of links for all algorithms in question was increased using additional WNICs on different channels. The percentage of links on the default (or common global) channel is lowest for MICA and DGA as shown in Figure 7.5.2b.

links between network nodes on different channels. Overall, the number of links in the resulting network topology has been increased by about 18 - 30% depending on the channel assignment algorithm. This matches also the percentage of links on the common global channel per algorithm as depicted in Figure 7.5.2b. For RAND, 86% of all links in the resulting network topology have been operated on the default channel in the median, 80% for DGA, and 74% for MICA.

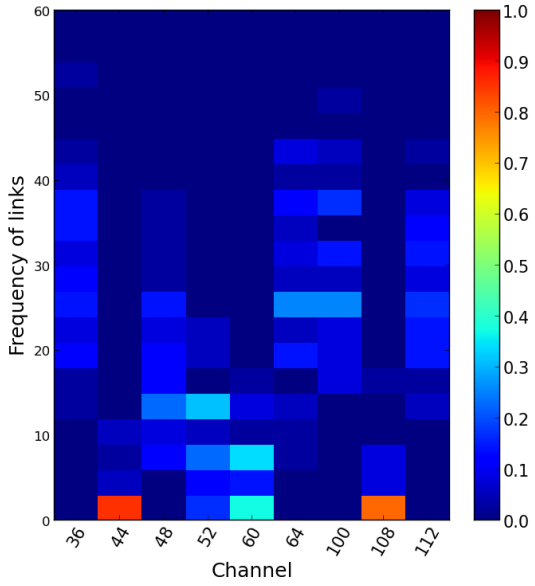
In summary, the network topology after the execution of MICA features the lowest percentage of links on the default channel. This can be credited to the fact, that nodes are able to leave the default channel in a channel switch, if no other links are affected in the process.

7.5.3 Channel distribution

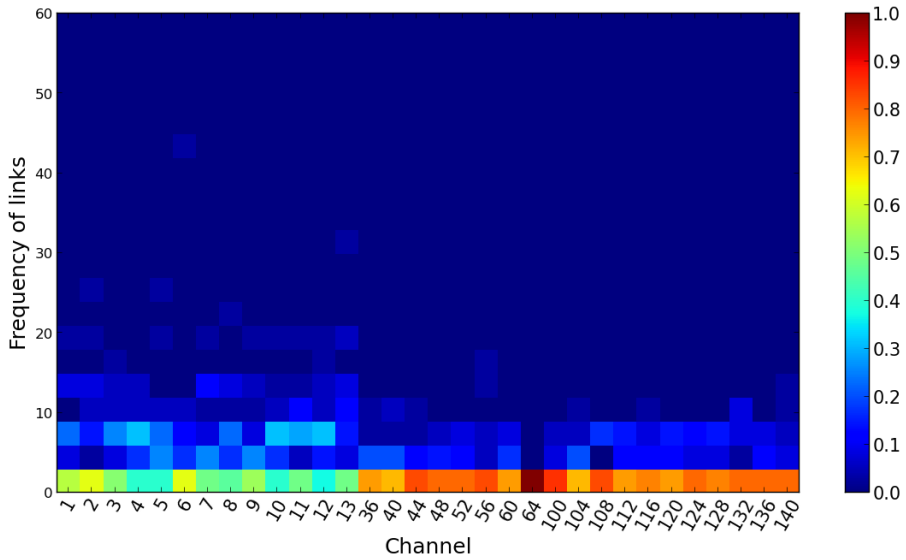
As next, we take a closer link on the channel distribution of all links in the network topology other than the ones operated on the default channel. The frequency of the remaining unidirectional links per channel over all experiment replications for MICA, DGA, and RAND is shown in Figure 7.5.3. For MICA and DGA, links have been established on all available channels and the frequency of most channels is in the same order. This can be credited to the random elements of the algorithms. For MICA, the order of the link-channel combinations (u, c) is randomly selected in each call of the `findMinimum` function. This way, when multiple channels would lead to the same decrease of interference, a random channel is chosen. For DGA, the initial channels for the switchable interfaces are selected randomly. Channel switches are restricted to those channels, that are already used by at least one direct neighbor in order to avoid dead interfaces. However, few channels have



(a) Channel distribution for MICA



(b) Channel distribution for DGA



(c) Channel distribution for RAND

Figure 7.5.3.: Channel distribution of links in the network topology that are not operated on the default channel for MICA, DGA, and RAND. The frequency of links per channel is derived from all 35 experiment replications.

not been used as much as others with DGA, in particular channels 44, 108, and 60. This can be credited to an implementation detail of DGA, in which the set of available channels is not randomized before finding the least interfering channel for an interface. This way, the order of channels is always the same, and channels with a lower index in the set of available channels are assigned more often if multiple channels achieve the same reduction of interference according to the used interference model.

An interesting observation can be made considering the channel distribution resulting

from the RAND algorithm. While there are also links established on all available channels, the amount of links using channels on the 2.4 GHz spectrum is higher than for the channels on the 5 GHz spectrum. This can be explained with the different signal range for the Atheros-based WNICs in the two different frequency bands as we have investigated in Section 3.3.1. Since we have not adjusted the transmit power depending on the utilized frequency band, the same network interface has a larger radio signal range when operated on 2.4 GHz compared to 5 GHz. This leads to a slightly higher frequency of links for the channels on the 2.4 GHz band compared to the 5 GHz band.

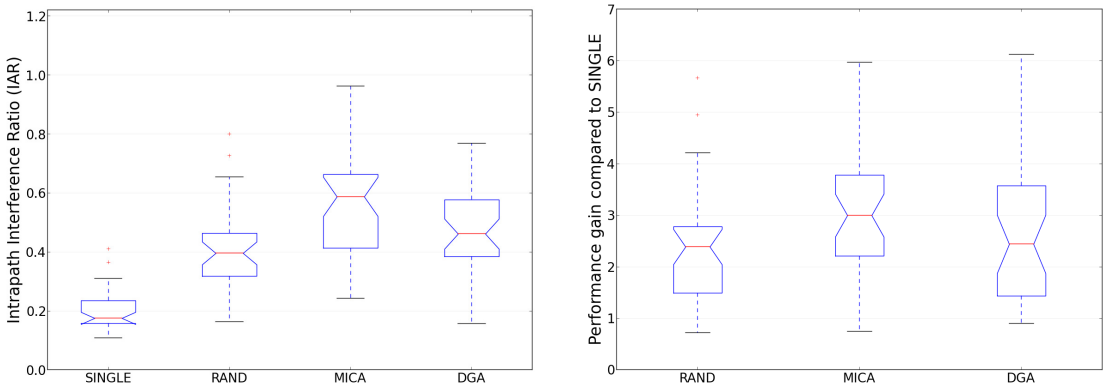
7.5.4 Intra-path Interference Ratio (IAR)

The results for 35 replications regarding the intra-path interference ratio (IAR) and the relative gain compared to the single channel network are shown in Figure 7.5.4. In each replication, we selected a 5-hop path randomly based on the WCETT routing metric [72]. We then measured the achievable throughput on all links sequentially and while all links have been activated simultaneously. The ratio of the two measurements is defined as IAR (see Section 4.3) and the results are shown in Figure 7.5.4a. Additionally, we compare the results of the single channel network (SINGLE) with the results of the channel algorithms in question by calculating the relative performance gain for the channel assignment algorithm as shown in Figure 7.5.4b.

For the single channel network, in the median of all experiment replications, only about 20% of the throughput has been preserved when all links are activated simultaneously compared to when activated sequentially. For the RAND algorithm about 40% of the throughput is preserved, which means that the achievable throughput compared to SINGLE has been doubled as depicted in Figure 7.5.4b. DGA preserves about 50% of the throughput, resulting in 2.5 times the throughput of SINGLE. MICA performs best with achieving almost 60% of the throughput when the links are activated simultaneously, thus being able to triple the throughput of the single channel network.

In summary, the results show that intra-path interference effects are extremely harmful to the network performance. In a single channel network, only 20% percent of the maximum throughput on a 5-hop path are achieved when the links are activated simultaneously. The results are partly caused by the setup of the DES-Nodes, and more specifically, with the small distance of the of the wireless network interfaces and antennas which results in board crosstalk and radiation leakage as described in Section 3.3.3. The experiments on *adjacent channel interference* (ACI) indicated severe performance degradation if multiple WNICs are operated on the same network node simultaneously on the same frequency band.

Compared to the single channel network, the channel assignment algorithms have been able to leverage the decrease in throughput of simultaneous transmissions significantly. Even the randomized approach (RAND) is able to double the throughput compared to the single channel network. DGA further increase the performance gain and MICA performs best, the achievable throughput triples compared to the single channel network. In con-



(a) Results for the intra-path interference metric (IAR).

(b) Performance gain compared to the single channel network.

Figure 7.5.4.: Results for the intra-path interference metric (IAR) and the relative performance gain compared to the single channel network.

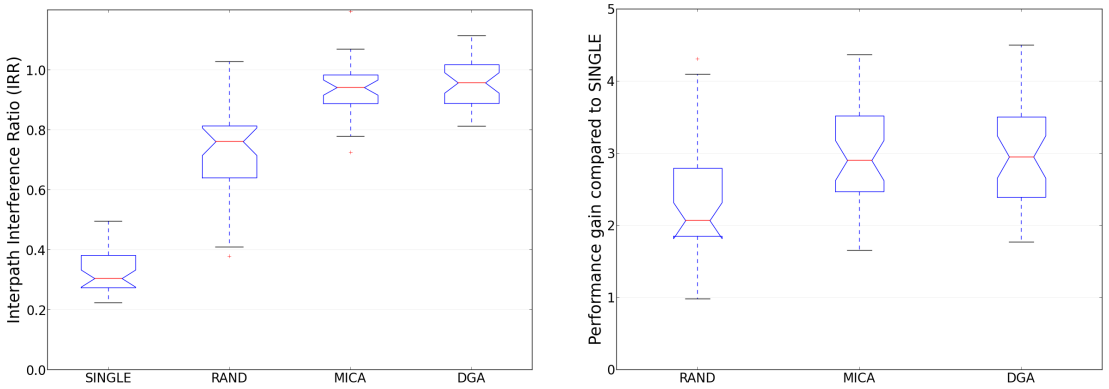
sequence, both channel assignment algorithms DGA and MICA significantly reduce the effects of intra-path interference.

7.5.5 Inter-path Interference Ratio (IRR)

The results for the inter-path interference performance metric (IRR) and the relative gain compared to the single channel network are depicted in Figure 7.5.5. In each replication, we selected a subset of 10 node-disjoint links of all nodes deployed in Arnimallee 6. We then measured the achievable throughput for all links sequentially and while all links have been activated simultaneously. The ratio of the two measurements is defined as IRR (see Section 4.3) and the results are shown in Figure 7.5.5a. Additionally, we compare the results of the single channel network (SINGLE) with the results of the channel algorithms in question by calculating the relative performance gain for the channel assignment algorithm as shown in Figure 7.5.5b.

For the single channel network, only 30% of the throughput could be preserved when all links are activated simultaneously. With RAND, about 75% percent of the throughput has been preserved, meaning that the achievable throughput in this scenario has been doubled compared to the single channel network. Both algorithms MICA and DGA perform best, preserving about 95% of the achievable throughput. Compared to the single channel network, the throughput was almost tripled with both algorithms.

The results show, that both algorithms MICA and DGA are efficiently reducing almost all effects resulting from inter-path interference in this particular scenario. Both algorithms are close to achieve the maximum throughput, thus eliminating all interference effects. The results are better than the ones for the IAR metric, since we only consider node-disjoint links in this scenario (see Section 4.3). This way, only one radio interface per node is activated throughout one experiment replication, which eliminates board crosstalk or radio leakage effects on a single DES-Node.



(a) Results for the inter-path interference metric (IRR).

(b) Performance gain compared to the single channel network.

Figure 7.5.5.: Results for the inter-path interference metric (IRR) and the relative performance gain compared to the single channel network.

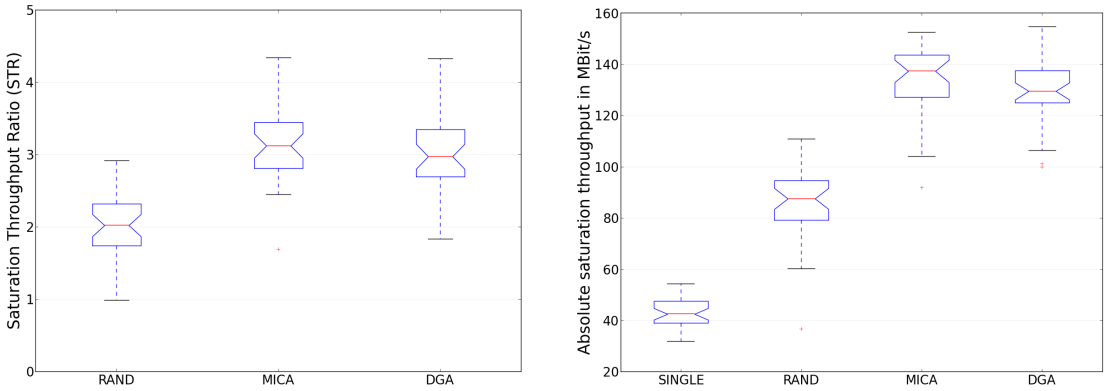
Compared to the single channel network, the channel assignment algorithms are capable to increase the throughput significantly in scenarios exposed to inter-path interference. The randomized approach (RAND) is able to double the throughput compared to the single channel network. DGA and MICA both triple the results of the single channel network and outperform RAND.

7.5.6 Saturation Throughput Ratio (STR)

The results for the saturation throughput ratio (STR) and the absolute network capacity are shown in Figure 7.5.6. To determine the network capacity, we selected 10 wireless links of all links in the testbed in the first step and measured the aggregate throughput of these links. For the next iteration we added another 10 links and repeated this procedure until the network was saturated or no more additional node-disjoint links could be found as described in Section 4.3.4. For all algorithms, at least 40 node-disjoint links have been found in all replications and the results for STR are shown in Figure 7.5.6a. Additionally, the absolute network capacity achieved by each algorithm is shown in Figure 7.5.6b.

For RAND, the network capacity has been doubled compared to the single channel network. With DGA the network capacity has been tripled, and MICA performs best with increasing the network capacity by 310%. A similar pattern is presented considering the absolute network capacity. The network capacity of the single channel network reaches 43 MBit/s. RAND achieves almost 90 MBit/s, DGA 130 MBit/s, and MICA performs best with almost 140 MBit/s.

In summary, the channel assignment algorithms are able to significantly increase the network capacity compared to the single channel network. While the randomized approach (RAND) doubles the network load compared to the single channel network, MICA and DGA achieve thrice the network load of the single channel network.



(a) Results for the saturation throughput ratio (STR).

(b) Absolute saturation throughput in MBit/s over 40 links.

Figure 7.5.6.: Results for the saturation throughput metric (STR) and the absolute saturation throughput considering 40 links.

7.5.7 Protocol Overhead (PO)

The results for the protocol overhead (PO) that is imposed on the network by the algorithm are shown in Figure 7.5.7. The plot shows the mean number of control messages sent per node during the channel assignment procedure over all experiment replications. With DGA, the mean number of messages sent by each network node is about 3300. The number of sent messages for MICA is significantly lower with only about 200 messages per node.

The reason for this gap can be found in the different communication protocols for channel switches. With MICA, the overhead is relatively low, since only the owner of a link can initiate a channel switch for this particular link. Additionally, only the two nodes adjacent to the link in question negotiate the channel switch with a three-way handshake. For a successful channel switch, only the *ChannelRequest* and *ChannelReply* messages are exchanged between the corresponding nodes, followed by possible *ChannelUpdate* messages.

DGA functions differently in that manner. In order to ensure convergence, the communication protocol involves all nodes of the 2-hop neighborhood of the node that initiates the channel switch procedure. The desired channel switch is announced to all nodes of the 2-hop neighborhood and all nodes must agree with a *ChannelAccept* message before the channel switch can be carried out. If at least one node sends a *ChannelReject* message, the procedure is aborted and a new round of the algorithm begins. Unsuccessful channel negotiations were common in the channel assignment procedure in this experimental study. The reason for this is two-fold. First, due to the dynamic network topology, the nodes do not always have complete and correct information of the interference set. If false assumptions for a pending channel switch are detected, the procedure is aborted. Additionally, a channel request is rejected, if another node is waiting to issue a channel request with a higher priority. Since the priority is only checked upon the reception of a channel request, many channel requests do not survive the conflict resolution phase.

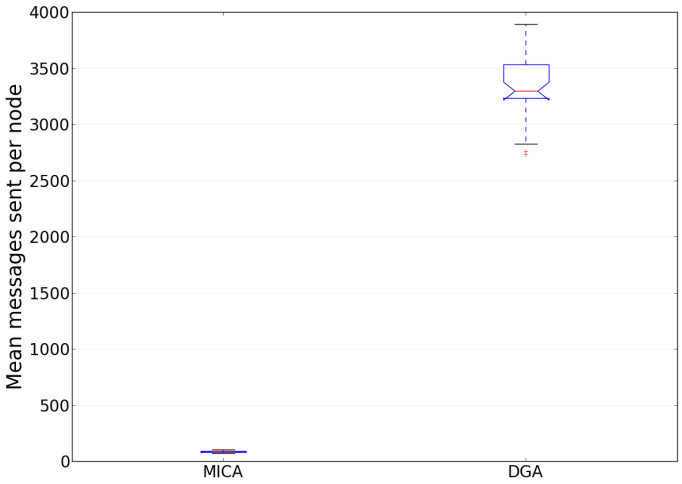


Figure 7.5.7.: Protocol overhead of MICA and DGA. The plot shows the mean number of sent messages per node over all experiment replications.

7.6 Impact of the interference model

For the results presented so far, both algorithms MICA and DGA have been configured to use the 2-hop interference model. We have shown in Chapter 5 that the *channel occupancy interference model* (COIM) determines interference relationships more accurately than the 2-hop interference model. For this reason, we investigate the impact of the utilized interference model on the performance gain in this last series of experiments by comparing the performance results of DGA utilizing the 2-hop and COIM interference model. As performance metric, we only consider the saturation throughput ratio (STR), since the metric gives an insight in the achievable network capacity.

First, we investigate the impact of the interference model on the interference set of the network nodes. The mean number of interferers for each network node in regard to the interference model is shown in Figure 7.6.1. With the 2-hop interference model, the mean number of nodes in the interference set is about 11, while it is 4 with COIM. The results are in line with the analysis of COIM as described in Section 5.4. The analysis revealed that the 2-hop interference model is too pessimistic in the DES-Testbed, since only about 40% of all 2-hop neighbors are actually interferers for a particular node.

The results according to the STR performance metric are shown in Figure 7.6.2. As also observed in Section 7.5.6, DGA with the 2-hop interference model increases the network capacity by about 290% compared to the single channel network. When DGA uses COIM instead, the network capacity is even further increased to 310%. These observations also hold for the absolute throughput measurements in Figure 7.6.2b. Thus, as expected, the channel assignment algorithm benefits from a higher accuracy of the interference model.

However, the increase of the network capacity is only marginal when utilizing COIM instead of the 2-hop model. The observations in the channel distribution in Section 7.5.3 showed, that DGA hardly assigns the last two channels in the set of the available channels.

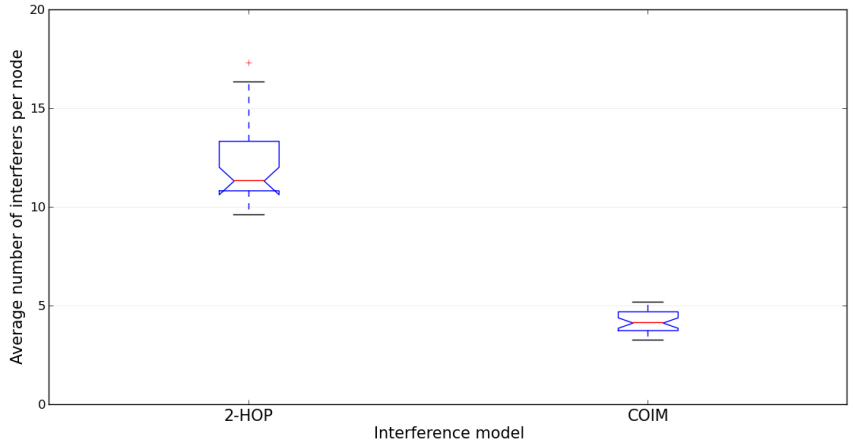
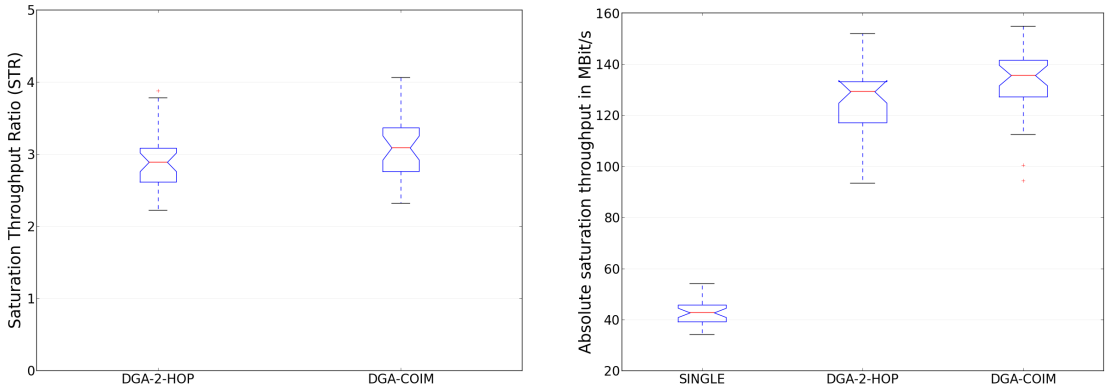


Figure 7.6.1.: Mean number of interferers according to the utilized interference model.



(a) Results for the saturation throughput ratio (STR).

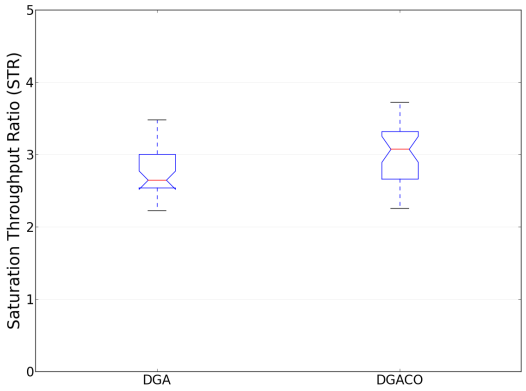
(b) Absolute saturation throughput in MBit/s over 40 links.

Figure 7.6.2.: Results for STR and the absolute saturation throughput according the utilized interference model with 10 available channels.

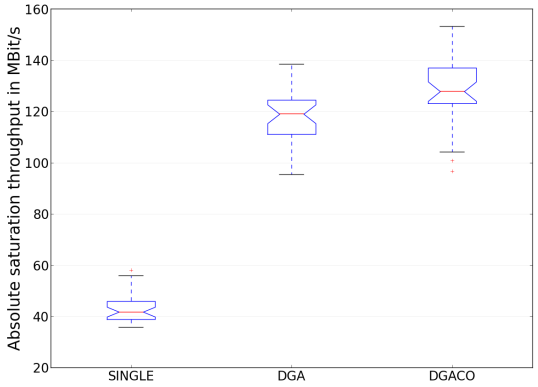
For this reason, the pessimistic 2-hop interference model may achieve good results albeit the high number of nodes in the interference sets. We assume that DGA utilizing COIM instead can cope better with a smaller number of available channels. Therefore, we repeat the experiment with a reduced channel set comprising only 6 channels instead of 10. The results for STR and the absolute throughput over 35 replications are shown in Figure 7.6.3.

The results for DGA with COIM are similar to the previous ones with 10 available channels. The network capacity could be increased by about 310% compared to the single channel network. When DGA utilizes the 2-Hop model, the network capacity could only be increased by about 260%. As expected, the impact of the reduced set of available channels is higher when DGA utilizes the 2-hop interference model compared to COIM.

In summary, the results show that channel assignment algorithms benefit from more accurate interference model, especially when the number of available channels is limited. With the trend of spectrum scarcity due to the increasing number of wireless devices in the unlicensed frequency bands, accurate interference models gain more importance.



(a) Results for the saturation throughput ratio (STR).



(b) Absolute saturation throughput in MBit/s over 40 links.

Figure 7.6.3.: Results for STR and the absolute saturation throughput according the utilized interference model with 6 available channels.

7.7 Discussion

In this chapter we presented a performance evaluation of distributed channel assignment algorithms in large-scale wireless mesh networks. Two existing approaches have been implemented based on DES-Chan and evaluated with the channel assignment benchmark (CAB). The performance evaluation showed that even a simple randomized strategy already delivers good results, being able to double the network capacity compared to the single channel network. Both algorithms MICA and DGA outperform the randomized approach, achieving thrice the network performance compared to the single channel network. MICA creates a slightly higher spectral diversity as DGA (see Section 7.5.3), which also reflects in the performance evaluation with slightly better results. The algorithms also differ in the protocol overhead they impose on the network. The number of control messages is significantly higher with DGA to ensure the convergence of the algorithm.

Furthermore, we have shown that the performance in regard to the network capacity could be further increased with the usage of accurate interference models. Compared to the 2-hop interference model, the performance results are better when COIM is utilized, especially in environments with a limited set of available channels.

In summary, the study has shown that channel assignment algorithms are capable of increasing the spectral diversity of multi-radio network and thus increasing the network performance in regard to metrics based on throughput. Both simple greedy strategies MICA and DGA deliver results in the same order and increase the network capacity by three times compared to the single channel network. However, both algorithms are not aware of external sources of interference. In the following chapter, we introduce our approach for external interference-aware channel assignment.

CHAPTER 8

External interference-aware channel assignment

With the pervasive use of wireless devices in our everyday lives, channel assignment algorithms need to solve a new challenge - they need to become aware of the radio activity of co-located networks and devices. This chapter describes such an algorithm that is capable to adapt to environments with interference caused by external devices. This chapter starts with a discussion on how spectrum sensing can be used to detect radio activity of external devices. We present DES-Sense, a spectrum sensing software tool for the commodity IEEE 802.11 wireless network interfaces of the DES-Testbed. Based on DES-Sense, an implementation candidate for an external interference-aware channel assignment algorithm is developed. The chapter closes with the results of the performance evaluation of the algorithm in the DES-Testbed and a comparison to DGA in multiple scenarios with different levels of interference.

8.1 Motivation

The increasing density of WLANs in urban areas leads to significant capacity limitations in the unlicensed 2.4 and 5 GHz bands [43]. Additionally, with the dawn of the *Internet of Things*, the number of wireless devices operating in the unlicensed frequency bands is dramatically increasing. It is envisioned, that by 2020, between 50 and 100 billion devices will be connected over the Internet [7]. The majority of these devices will communicate using a local Access Point or Gateway to connect to the Internet. Many of these devices will be equipped with radios operating in the unlicensed frequency spectrum. Thus, solving the challenge of spectrum scarcity is crucial for efficient wireless communications. From the point of view of the network operator, co-deployed WLAN networks as well as devices based on other technologies such as Bluetooth [45] or ZigBee [44], can be considered *external*, since they are not under control of the network operator. Thus, we summarize their activity with the term *external interference* in the remainder of this chapter.

The rising congestion of the wireless radio channels poses crucial challenges for the field of channel assignment. To exploit the benefits of spectral diversity, the channel assignment approaches have to consider not only the network-wide *channel congestion*[1] but also the local channel conditions which may be affected by co-located devices. Otherwise, it may happen that channels are assigned, which are already heavily congested by the radio activity of external devices. Therefore, channel assignment approaches have to be *external interference-aware* in exposed environments.

Several challenges have to be solved for the implementation of an external interference-aware channel assignment. The contribution of this chapter are the following

- A measurement-based method is developed to assess the local radio channel congestion. The DES-Chan framework has been extended to provide the data of the channel congestion to the channel assignment algorithms.
- An *external interference-aware channel assignment* (EICA) algorithm is developed using the statistics of the local channel congestion.
- A performance evaluation in the DES-Testbed is carried out to investigate the efficiency of EICA in scenarios with different levels of interference.

At first, the approach to measure the congestion levels of the wireless radio channels in a distributed manner is presented. Then, EICA is introduced and the chapter finishes with the performance evaluation of EICA in the DES-Testbed.

8.2 Detecting external radio activity

8.2.1 Challenges

A crucial prerequisite for external interference-aware channel assignment is that the network nodes have to be aware of ongoing local radio activity. This can be done passively, either by detecting external devices and monitoring their radio activity or by measuring the congestion level of the available radio channels. The main challenges to detect the radio activity of external devices are the following:

- The location of external devices and patterns of their radio activity are usually not known before the network operation.
- A wide range of different devices based on different technologies operate in the unlicensed frequency spectrum. These devices can range from co-located WLAN Access Points and clients to Bluetooth and ZigBee devices, wireless keyboards, game controllers, and many more.

[1] Often used synonyms for this term in the literature are *channel load*, *channel occupancy*, and *channel busy fraction*.

- The traffic patterns of the external devices are not known and may be dynamic over time. For instance Access Points of co-located WLAN networks may send beacons periodically, while mobile devices such as smartphones connected to these Access Points roam the network environment only in the day time.

For these reasons the detection of all external wireless devices in the network environment requires much effort. One solution to measure the local radio activity is to determine the congestion levels of the available radio channels, thus knowing which channels are already heavily used. Commercial spectrum analyzers, such as WiSpy [135] and Spectrum XT [136], are hardware systems for this task that allow the detection of RF activity in the unlicensed 2.4 and 5 GHz spectrum. Alternatively, software solutions using off-the-shelf IEEE 802.11 hardware can be used for the same purpose. This approach has the advantage that no additional hardware is required and it can be usually integrated into existing systems. Measuring the channel load has the benefits that the channel conditions are taken into account independently of the particular devices causing the channel congestion.

8.2.2 Measuring channel congestion

Measuring the *channel congestion* can be done in multiple ways. The standard extension IEEE 802.11k for radio resource measurements and management [137] introduces methods for interference measurements derived from the channel congestion [138]. Based on these measurements, *channel load* and *medium sensing time histogram* reports are exchanged among the network nodes. With these reports, clients connect to Access Points considering not only the signal strength but also the congestion level of the channel utilized by the Access Points. The mechanisms were introduced to distribute the network traffic more evenly across the available Access Points and radio channels.

Another method to measure the channel congestion is to directly access the *carrier sensing* statistics of the wireless network interface. Atheros-based WNICs, as used in the DES-Testbed, maintain two registers that can be used for this purpose [139, 140]. The *cycle time* (or *time slot*) counter is incremented with every clock tick and depends on the used clock for the particular Atheros chipset. The *medium busy counter* holds the information for how many cycles the medium has been sensed busy in regard to the *clear channel assessment* (CCA) function. Current drivers for the Atheros-based WNICs, such as `madwifi` [74] and `ath5k` [85] expose these statistics. Calculating the ratio of these two values results in the measured channel congestion for the captured time period.

The method of measuring the channel congestion based on carrier sensing has already been used for many diverse protocols and application scenarios. These applications include interference-aware routing metrics [141], rate adaptation mechanisms for IEEE 802.11 [140, 142], resource allocation [143], and localization with ranging for IEEE 802.11 devices [144]. Also, as part of *cognitive radio* research, *spectrum sensing* and *dynamic spectrum access* (DSA) are introduced mechanisms to detect white spaces to counter the radio spectrum scarcity problem [130].

Value	Description
channel_time	Holds the number of times lots the radio has been operating on the particular channel.
channel_time_busy	Holds the number of time slots in which the medium has been sensed busy.
channel_time_rx	Holds the number of time slots in which the radio was receiving a transmission.
channel_time_tx	Holds the number of time slots in which the radio was transmitting a frame.

Table 8.2.1.: Overview of Atheros AR5413 carrier sensing registers.

8.2.3 DES-Sense

The sensing component DES-Sense allows real-time measurements of the channel congestion in the DES-Testbed. DES-Sense calculates the channel congestion based on the statistics of the carrier sensing registers of the wireless network interface as described above. As next, we present the architecture of DES-Sense and show results of measuring the channel congestion in the environment of the DES-Testbed, which is co-located to several WLANs on the campus of the Freie Universität Berlin.

Implementation

DES-Sense measures the channel congestion based on the *carrier sensing* statistics of the wireless network interface. We retrieve the statistics by periodically reading the corresponding 32-bit registers of the wireless network interfaces with the AR5413 chipset. The `ath5k` driver [85] provides a wrapper function which returns the values given in Table 8.2.1. In order to calculate the channel congestion, only the values for the `channel_time` and `channel_time_busy` are required. With these values we can calculate the channel congestion CG_c for channel c with

$$CG_c = \frac{channel\_time\_busy}{channel\_time} \tag{8.1}$$

The result for the channel congestion is a value between 0 and 1, describing what fraction of the measurement interval the medium was sensed busy. A value $CG_c = 0.5$ corresponds to a channel load of 50% for channel c in the measurement interval.

Challenges

In order to use this method for measuring the channel congestion efficiently, several challenges have to be solved. First, the measurements of the channel congestion on the channels in question are performed in real-time. Depending on the radio activity patterns of the local devices, the channel congestion may be very dynamic over time, thus requiring repetitive measurements to maintain correct information. With multi-radio mesh nodes, this

can either be solved with a dedicated interface, that permanently runs the measurement task for the channels in question. The drawback of this approach is that the dedicated interface can not be utilized for data transmissions. Another method is to perform the monitoring measurements event-based, in case a change of the link quality is observed, for example, when the throughput on that particular link drops. With this method, a traffic flow must be stopped for the duration of the measurement phase and can only be resumed after a less congested channel is found and switched to. This will lead to a higher delay for this particular flow.

Another challenge results from the fact, that a wireless interface can measure the channel congestion for only one channel at a time. This means, that simultaneous measurements of all available channels are not possible with one wireless interface. With the set of available channels K and the measurement duration t_c for each channel $c \in K$, the duration for a complete measurement phase is

$$T_M = \sum_{c \in K} t_c \tag{8.2}$$

Finally, from the carrier sensing statistics alone, it can only be derived that the channel has been utilized, but not by which station. Therefore, based on the statistics we can not distinguish traffic from our own network from the radio activity of external networks and devices. To solve this problem, the monitoring interface can be set in *monitor mode* and thus capture and analyze all received frames during the measurement interval. This way, we can identify frames sent by nodes from our network and distinguish the corresponding channel load from external radio activity.

Architecture

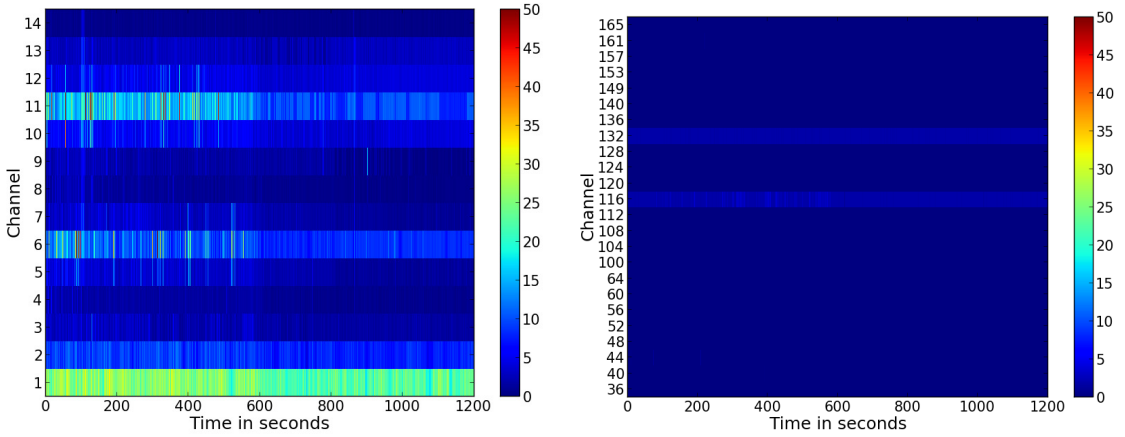
DES-Sense has been integrated into the DES-Chan framework and consists of the following two components:

- *Sensing component* - The sensing component is a daemon that periodically retrieves statistics about the channel congestion as described above. The sensing component can be configured dynamically with the set of channels $C = \{c_1, c_2, .., c_k\}$ that will be monitored and the duration $T = \{t_1, t_2, .., t_k\}$, each channel is monitored.

- *IPC interface* - An *inter process communication* (IPC) interface is provided that allows algorithms to retrieve the channel congestion statistics. The algorithms can query the service via the interface to update their channel congestion statistics to fuse the information into their channel assignment decision.

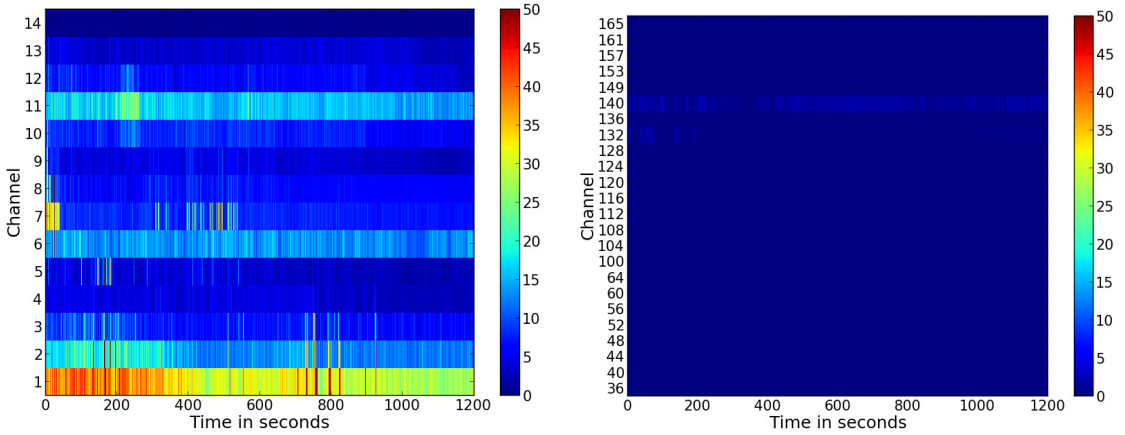
8.2.4 Experimental evaluation

The DES-Testbed is deployed on the premises of the computer science campus of the Freie Universität Berlin. Throughout the deployment area, Access Points of co-located WLAN networks are distributed to provide Internet access to students and research staff.



(a) Indoor node a6-011 at 2.4GHz

(b) Indoor node a6-011 at 5GHz



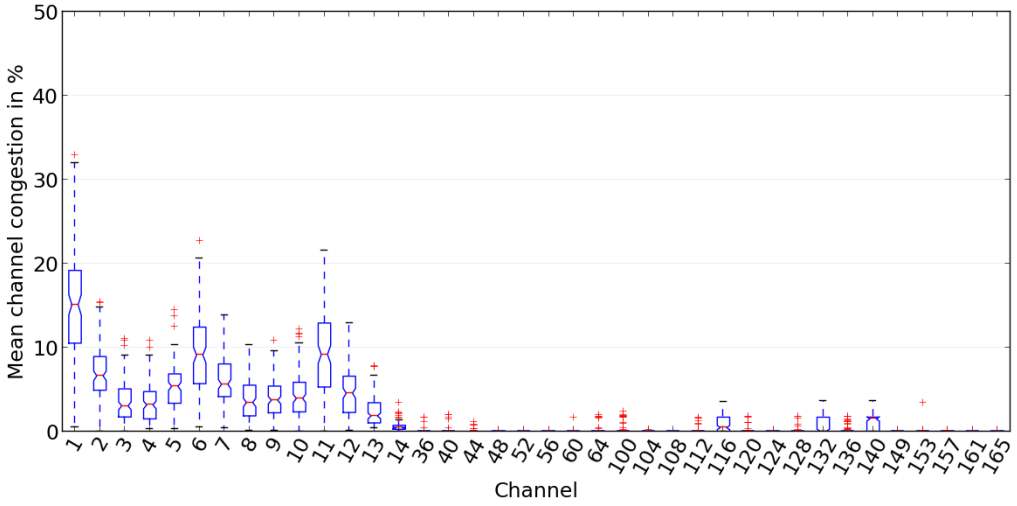
(c) Outdoor node t9-ext-154 at 2.4GHz

(d) Outdoor node t9-ext-154 at 5GHz

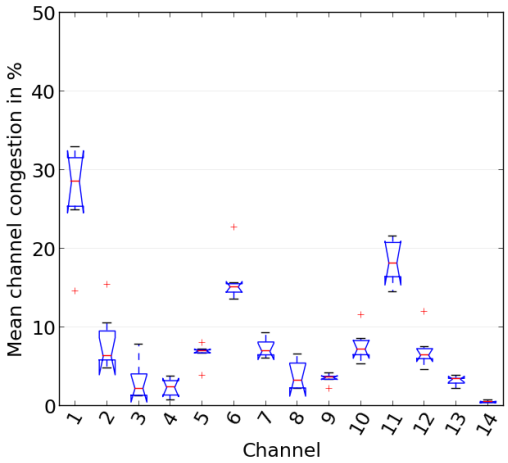
Figure 8.2.1.: Exemplary results of channel congestion measurements in the environment of the DES-Testbed. A significant channel congestion is detected on selected channels in the 2.4 GHz spectrum. The channels on the 5 GHz spectrum are not utilized.

In this experimental study, we measure the basic channel congestion in the environment of the DES-Testbed while the nodes of the testbed are completely silent. We activated one wireless interface on each testbed node and measured the channel congestion for all available channels on the 2.4 and 5 GHz frequency band for 20 minutes each. Exemplary results for the measured channel congestion for one indoor and outdoor node for both frequency bands are depicted in Figure 8.2.1.

The results reveal two interesting characteristics of the environment of the DES-Testbed. First, the available channels on the 5 GHz spectrum are hardly utilized in the environment of the DES-Testbed. Second, a significant channel congestion of up to 40% has been detected on selected channels on the 2.4 GHz band. The channels with the highest congestion are 1, 6, 11 and we can see that the adjacent channels suffer from *adjacent channel interference* (ACI) since they overlap. Further investigation revealed that the Access Points of the infrastructure WLAN of the university is operating on exactly these channels. Most



(a) Mean channel congestion over all network nodes.

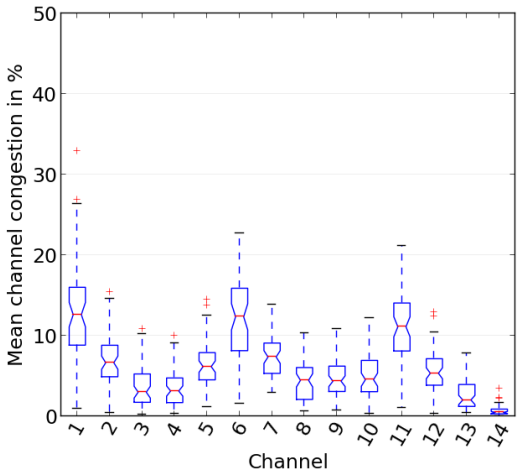


(b) Mean channel congestion over all outdoor nodes.

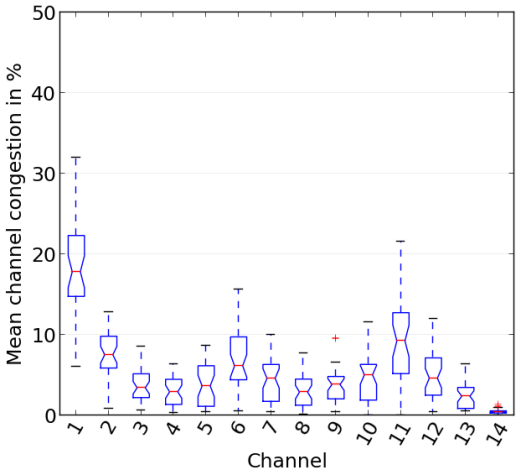
Figure 8.2.2.: Mean channel congestion measured in the DES-Testbed with DES-Sense. In (a), the mean channel congestion over all nodes for all channels of the 2.4 and 5 GHz band are depicted. In (b), the channel congestion as measured by the outdoor nodes for the 2.4 GHz band is shown.

of the frames that caused the channel congestion were beaconing frames sent by Access Points. The results show that the channel congestion is a feasible mechanism to detect co-located WLAN networks and their utilized channels. We can also see, that the channel congestion measured at the outdoor node is higher on all channels in the 2.4 GHz band than for the indoor node. This can be credited to the outdoor nodes being more exposed than the indoor nodes. The same effect can be observed in the network topology of the DES-Testbed. The node degree of the outdoor nodes is usually much higher than the degree of the indoor nodes.

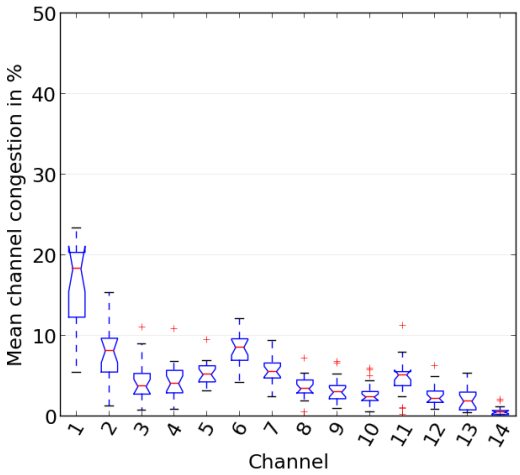
The mean channel congestion over all nodes and channels is depicted in Figure 8.2.2a. The plot shows the same pattern with a significant channel congestion in the 2.4 GHz



(a) Channel congestion at Takustraße 9.



(b) Channel congestion at Arnimallee 6.



(c) Channel congestion at Arnimallee 7.

Figure 8.2.3.: Comparison of the channel congestion per building of the DES-Testbed. The pattern of the peaks for the channels 1, 6, 11 is visible in all three plots. However, the congestion of channels 6 and 11 is lower in the buildings of Arnimallee compared to Takustraße.

spectrum and hardly any radio activity in the 5 GHz spectrum. Figure 8.2.2b shows the mean channel congestion over all measurements from the outdoor nodes in the 2.4 GHz band. As expected, the channel congestion at the outdoor nodes is higher than the over all mean values, because the outdoor nodes are more exposed than the indoor nodes.

As next, we investigate the channel congestion levels in the different buildings the DES-Testbed is deployed in. The mean channel congestion for the nodes in the buildings Takustraße 9, Arnimallee 6, and Arnimallee 7 are depicted in Figure 8.2.3. In all plots, the pattern of the peaks for the channel congestion for channels 1, 6, and 11 is visible. However, the congestion for channels 6 and 11 is lower in the buildings in Arnimallee than in Takustraße. Thus, the placement of a particular node has a great impact on the measured channel congestion levels.

In summary, the experimental evaluation has shown that the method for measuring the channel congestion works. The results of the measurements comply with the channel usage of the co-located Access Points of the university WLAN infrastructure. Furthermore, the evaluation revealed two interesting characteristics of the radio environment of the DES-Testbed. First, there is hardly any activity in the 5 GHz spectrum. We can treat the channels on these bands as almost interference-free. Second, a significant channel congestion has been measured by almost all nodes on the channels 1, 6, 11. Therefore, these channels should be avoided in our experiments to avoid instability due to effects of external interference.

8.3 External Interference-Aware Channel Assignment (EICA)

8.3.1 Idea

The *external interference-aware channel assignment* (EICA) algorithm incorporates the channel congestion information provided by DES-Sense to avoid already heavily congested channels in the local environment of the network nodes. EICA results from an extension of the *distributed greedy algorithm* (DGA) [48], which in the original version only considers intra-network interference and not external interference.

EICA extends DGA with the presented procedure to measure the channel congestion on each network node. Based on these measurement data, a blacklisting mechanism is introduced, that removes channels with a high load from the set of the available channels.

8.3.2 Algorithm

With DGA, the least used channels in the *interference set* of a node are assigned to its network interfaces. The interference set S_n of a node n consists of all nodes and their channel assignment whose transmissions affect sending and receiving at node n. To preserve the network connectivity, one interface on every node is operating on a global common channel (or default channel) and is not used for channel assignment.

The EICA algorithm, including the channel congestion measurement procedure (lines 1–6) and the channel selection procedure (lines 7–15), is given in Algorithm 4. At the network initialization, all nodes measure the local channel congestion on all available channels K. If the channel congestion CG_c at a particular node exceeds a threshold T_{CG} for channel $c \in K$, the channel c is removed from the set of the available channels and not considered for the channel assignment on this node. More formally, the set of available channels K after measuring the channel congestion is updated as follows

$$K \leftarrow \{\forall\, c \in K \mid CG_c < T_{CG}\} \tag{8.3}$$

In other words, if the local congestion on channel c exceeds the threshold T_{CG}, a negative impact on the performance of links operating on this channel is expected and thus, the

channel should be avoided. However, a channel can only be removed from the set if the number of available channels is still higher than the number of switchable interfaces of the node. Thus, it is ensured that each interface can be operated on a different channel.

It is important to note that the channel congestion is measured in a distributed manner on all nodes. The channel congestion may vary throughout the environment of the network, especially in large-scale wireless networks that cover a large area and possibly multiple buildings. Thus, channels should not be avoided globally, but only in the areas in which the network nodes have detected a significant load on the particular channels.

After the channel congestion measurement, the network is initialized with setting one interface to the default channel. After retrieving the local communication graph based on the ETX metric [100], the interference set S_n of each node is determined according to the utilized interference model. With an *interference cost function* f_I as given in Equation (8.4), the spectral overlap of two channels is used to determine the level of interference in the interference set. The cost function takes the center frequencies f_1 and f_2 of two radio channels and the additional parameter α into account. α denotes the minimum frequency difference of orthogonal channels. A value of $f_I = 0$ states that the channels are orthogonal, meaning they do not interfere with each other.

$$f_I(f_1, f_2, \alpha) = max(0, \alpha - |f_1 - f_2|) \tag{8.4}$$

Using the interference cost function, each node tries to assign the channel that results in the lowest level of interference in its interference set to its interfaces (except the default interface) according to the channel selection procedure (lines 7–15) in Algorithm 4. In each iteration, the algorithm considers exactly one interface and calculates the interference cost for all available channels K with the current channel assignment of the interference set. The result is the channel c_{best} with the lowest overall interference cost. If c_{best} is different than the currently assigned channel for this interface, a channel switching procedure is initiated. As an additional constraint, only channels can be assigned that are used by at least one neighboring node, in order to avoid isolated interfaces.

The channel switching procedure is based on a 3-way handshake [133]. It is initiated by sending a `ChannelRequest` message to all nodes in the interference set S_n. When receiving a `ChannelRequest` message, a node puts a possible own channel switch intent on hold. The recipients of the `ChannelRequest` reply with a `ChannelAccept` message, if the proposed channel switch does not conflict with their own intended channel switches. The originator of the request applies the channel switch, if all nodes of the interference set reply with a `ChannelAccept` message. A following `ChannelUpdate` message is sent to the nodes in the interference set to confirm the channel switch. If the `ChannelRequest` conflicts with another pending channel switch, a node replies with a `ChannelReject` message. When receiving a `ChannelReject` message, the originator of the request aborts the current channel switching procedure by sending a `ChannelAbort` message to all nodes in its interference set. The algorithm terminates, when the local interference can not be reduced any further.

Algorithm 4 Channel congestion measurements and channel selection with EICA

Require: $S_n \leftarrow$ Interference set of node n
$K \leftarrow$ Set of available channels
$c_j \leftarrow$ channels assigned to each node $j \in S_n$
$c_i \leftarrow$ current channel of the interface i
1: **for** k **in** K **do**
2: $CG_k \leftarrow measure\_channel\_congestion(k)$
3: **if** $CG_k > T_{CG}$ **then**
4: $K \leftarrow K \setminus k$
5: **end if**
6: **end for**
7: $F_{best} \leftarrow \sum_{j \in S_n} f_I(\alpha, c_i, c_j)$
8: $c_{best} \leftarrow c_i$
9: **for** k **in** K **do**
10: $F_k \leftarrow \sum_{j \in S_n} f_I(\alpha, k, c_j)$
11: **if** $F_k < F_{best}$ **then**
12: $F_{best} \leftarrow F_k$
13: $c_{best} \leftarrow k$
14: **end if**
15: **end for**

8.3.3 Implementation

The EICA implementation is based on the DGA prototype that has been developed in Section 7.3. The *EICA* class comprises the channel congestion measurement phase and the channel selection algorithm. The *Messaging* class contains the implementation of the communication protocol for channel switch synchronization. At the network initialization, all nodes measure the local channel congestion with DES-Sense on all available channels K. Afterwards, one interface on each network node is set the default channel while the switchable interfaces are initialized by selecting a channel randomly from the set of the remaining available channels. As next, information about the local 2-hop neighborhood of each node is retrieved with the *Neighborhood Discovery* service of the DES-Chan framework using the ETX metric [100]. Once the local topology has been retrieved, the interference set is created by querying all 2-hop neighbors for information about their current channel assignment. This completes the initialization after which the program flow is determined as depicted in the flow diagram in Figure 8.3.1.

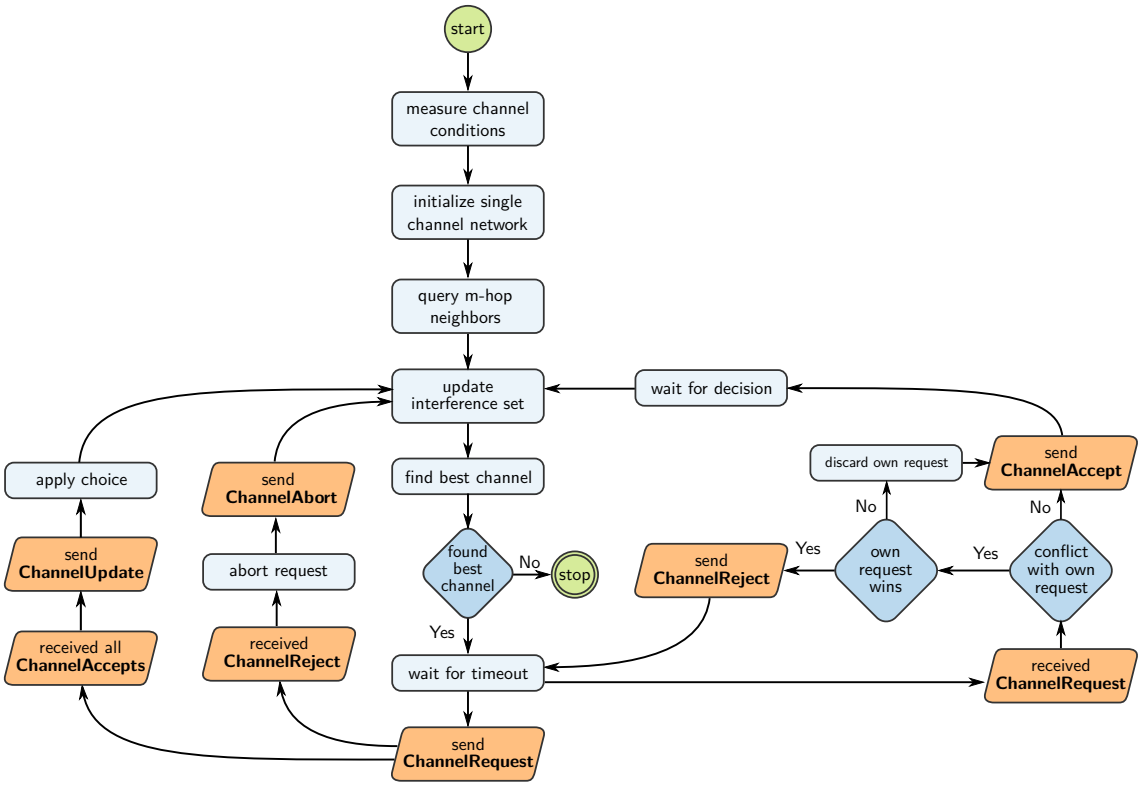


Figure 8.3.1.: Flowchart of the distributed EICA implementation. The algorithm is based on DGA, an additional phase to measure the channel congestion has been introduced when the algorithm is started. Methods of the *EICA* class are colored light-blue, receiving and sending messages are represented by light-orange parallelograms.

8.4 Performance evaluation

The performance evaluation in this chapter is motivated by the following two questions.

- How harmful is external interference for the performance of wireless networks if channel assignment algorithms do not consider such interference?

- How well can EICA cope with high levels of interference in the network environment?

In this study, we compare the performance of the channel assignment algorithms DGA and EICA in scenarios with varying levels of interference. Throughout the study, we use a subset of the network nodes of the DES-Testbed to act as co-located interferers in order to create diverse interference levels. As performance metric, we consider the *external interference ratio* (EXR), which is described in Section 4.3.1 and briefly revisited in the course of this chapter. Additionally, we investigate the interference levels of the different scenarios and the impact of the blacklisting mechanism on the set of available channels.

Parameter	Value
Number of nodes	126 DES-Nodes each equipped with one IEEE 802.11b/g radio and two IEEE 802.11a/b/g radios.
Algorithms	DGA, EICA
Number of interferers	10, 30, 50
Interference traffic pattern	1CBR, BURST, 6CBR
Default interface / channel	wlan0 / 14
Available channels	36, 44, 48, 52, 60, 64, 100, 108, 112
Threshold T_{CG}	0.10
Performance metric	EXR
Interference model	2-hop model
Replications	35 for each scenario

Table 8.4.1.: Experiment settings for the performance evaluation of EICA.

8.4.1 Experiment design

The settings for this experimental study are defined in Table 8.4.1. We use 126 network nodes of the DES-Testbed for the experiments. For the channel assignment, all three IEEE 802.11 interfaces are used, set to ad-hoc mode and 6 MBit/s data rate. Channel 14 is used for the common global channel since measurements have shown that this particular channel is not utilized in the environment of the DES-Testbed. In all experiments, a subset of these nodes act as interferers (varying from 10, 30, and 50 nodes), while the remaining network nodes run the channel assignment. The interferers are chosen randomly from all network nodes in the beginning of each replication of an experiment. A node selected as interferer will generate traffic throughout the experiment using two of its WNICs tuned to random channels of the available channel set (see Table 8.4.1).

The interfering nodes use three different traffic patterns: with 1CBR the interferers generate 1MBit/s *constant bit rate* (CBR) traffic. With 6CBR, the interferers generate 6 MBit/s CBR traffic, resulting in that the interferers jam the particular channels since the data rate of the interfaces is set to 6 MBit/s as well. For the last traffic pattern, the interferers replay a bursty traffic schedule (BURST) that has been recorded before in the testbed of single hop TCP traffic. The average data rate is roughly 2 MBit/s, the delay between subsequent bursts is chosen randomly and may exceed the duration of the channel congestion measurement procedure for a single channel. The interfering nodes are activated in the beginning of the EICA execution, thus, the increased channel load can be detected during the execution of the algorithm.

The three groups of interferers combined with the three different traffic patterns result in a total of 9 scenarios. We run 35 experiment replication for all 9 scenarios, resulting in a total of 315 experiments with a total runtime of the study of about 13 days.

8.4.2 Performance metric

In each scenario, we measure the *external interference ratio* (EXR) performance metric with the following steps. First, the specified number of interferers are randomly selected of all available network nodes. At the beginning, the interferers start generating traffic while all other nodes run the channel assignment procedure. After the channel assignment procedure has finished, the evaluation phase is started to measure the network capacity. For this, we select all wireless links L with the constraint, that each node is at most incident to not more than one link in L. This gives us the maximum number of node-disjoint single hop links in the network. Then, we measure the throughput of all links in L when they are activated simultaneously with 6 MBit/s UDP traffic. The aggregate throughput over all links expresses the network capacity and can be written as

$$T_{noif} = \sum_{l \in L} t_l^{noif} \tag{8.5}$$

We repeat this measurement, but this time, the interferers are actively transmitting messages throughout the measurement. The aggregated throughput of the second measurements is defined as

$$T_{if} = \sum_{l \in L} t_l^{if} \tag{8.6}$$

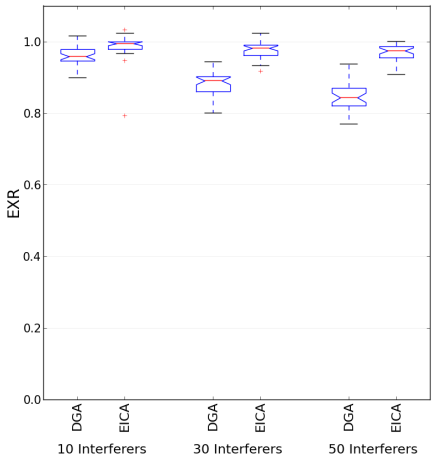
The EXR metric is then defined as the ratio between the aggregated throughput while the interferers are active and when they are not, which is

$$EXR = \frac{T_{if}}{T_{noif}} \tag{8.7}$$

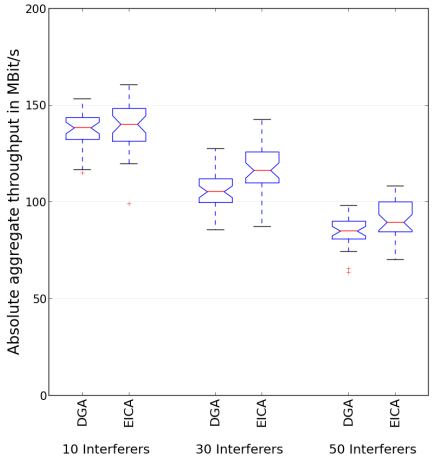
A value for EXR close to 1 means that the aggregate throughput is not reduced when the interferers are active compared to when they are not. In this case, the network capacity is not affected by the external interference. A lower value for EXR means that the transmissions of the interferers have decreased the network capacity.

8.4.3 Results

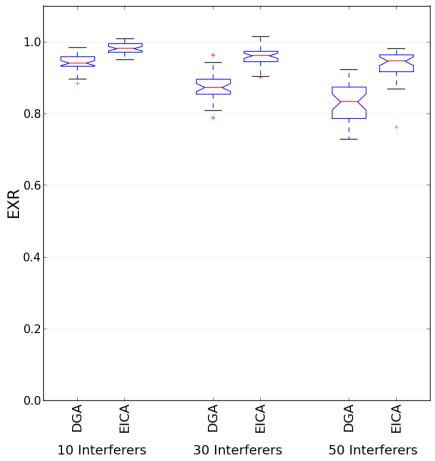
The results for the scenarios with the 1CBR traffic pattern for the EXR metric are depicted in Figure 8.4.1a. As expected, with the growing number of external interferers, the EXR decreases for DGA since the algorithm does not consider external interference. With 50 interferer, the network capacity drops by about 15% compared to when the external interferers are not activated. The plot shows that with EICA the network capacity is hardly affected by the external interference. The network capacity drops only by 3% compared to when the interferer are not active in the median of all 35 experiment replications with 50 interferers. The absolute value for the aggregate throughput when the interferers are active are shown in Figure 8.4.1b. With the increasing number of interferers, the absolute



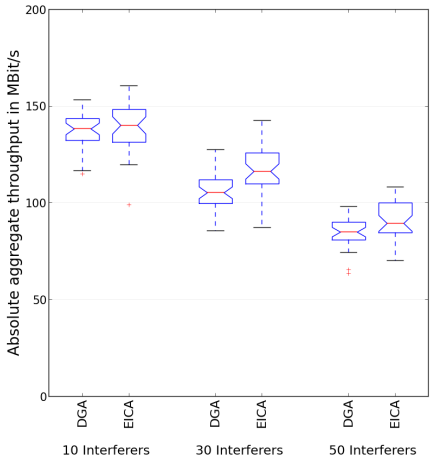
(a) EXR results for 1CBR.



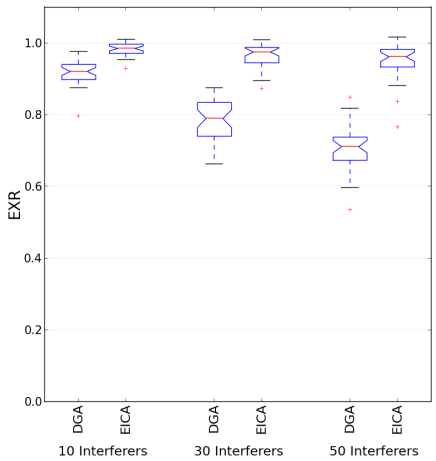
(b) Throughput results for 1CBR.



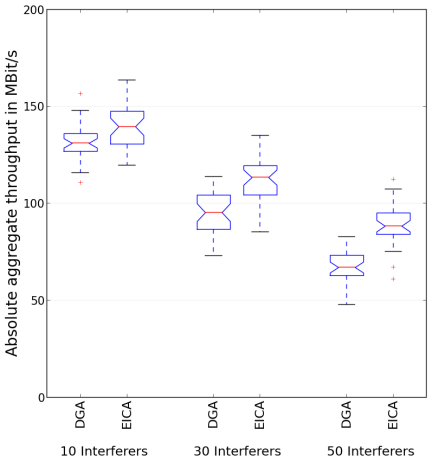
(c) EXR results for BURST.



(d) Throughput results for BURST.



(e) EXR results for 6CBR.



(f) Throughput results for 6CBR.

Figure 8.4.1.: Results of DGA and EICA. In (a) and (b), the results are shown for the 1CBR traffic pattern. The results for the BURST traffic pattern are shown in (c) and (d), and for the 6CBR traffic pattern in (e) and (f).

throughput decreases, since more nodes act as interferers and fewer nodes run the channel assignment. As expected, the results for the aggregate throughput for EICA with about 90 MBit/s are higher than the results for DGA with about 85 MBit/s.

The results for the BURST traffic pattern are depicted in Figure 8.4.1c. Similar to the first set of results, the EXR decreases for both algorithms with the growing number of interferers. The results for EICA are slightly worse compared to the 1CBR scenario - with 50 interferer the network capacity drops by about 6% in the median while with DGA the network capacity drops by 15%. This can be explained by the bursty traffic pattern, which makes it harder to capture the external interference in the channel congestion measurement phase. It may happen, that while the channel state is measured, the interfering node is currently not sending, thus it is not captured. The absolute value for the aggregate throughput when the interferers are active are shown in Figure 8.4.1d. With 50 interferers, EICA achieves an aggregate throughput of about 87 MBit/s while DGA achieves 78 MBit/s.

The results for the increased interferer activity by using 6CBR traffic are depicted in Figure 8.4.1e. As expected, the higher level of external interference compared to the previous scenarios has an impact on the network capacity of both algorithms. The impact is again higher on DGA, the network capacity drops by 30% percent compared to when the interferers are not active. With EICA, the network capacity only drops by 5%. The results for the aggregate throughput as depicted in Figure 8.4.1f show a similar trend. While EICA achieves about 88 MBit/s, DGA achieves only about 66 MBit/s.

Over all 9 scenarios, the experimental performance evaluation shows that EICA outperforms DGA. This is as expected since DGA does not consider external interference in the channel assignment. Throughout all scenarios, the results for EICA regarding network capacity are only 6% less than without the interfering nodes being activated. Thus, EICA is capable to detect channels with a high load and to avoid these channels for the channel assignment. However, the negative impact on the network capacity is strong for algorithms like DGA that do not consider external interference.

Channel congestion measurements

As next, we attempt to quantify the interference level of the environment created by the interfering nodes in regard to the traffic pattern and the amount of external interferers. For this, we take a look at the mean channel congestion of all available channels as measured with EICA during the channel congestion measurement procedure. The available channels are all in the 5 GHz frequency band, where preceding measurements have shown no activity of testbed-external devices. Thus, the measured channel congestion results from the activity of the interferers as selected for each scenario.

The results are depicted in Figure 8.4.2. As expected, for all three traffic patterns the measured channel load increases with the growing number of interferers. Also, the more traffic the interferers generate, the higher is the measured channel congestion. Thus, as expected, the channel congestion is highest for the 6CBR traffic pattern. The mean channel

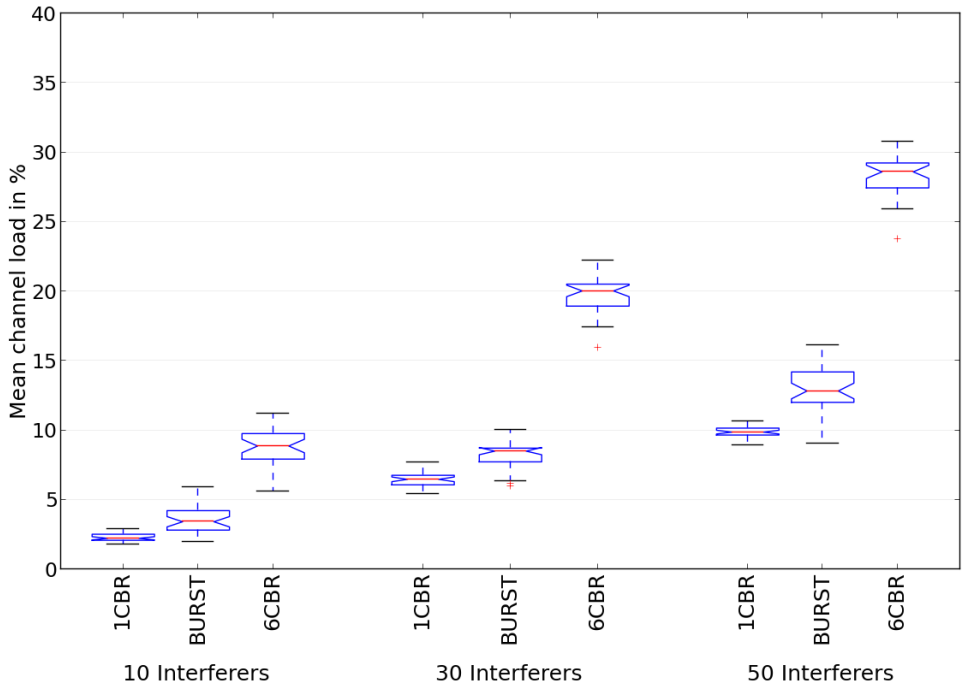


Figure 8.4.2.: Mean measured channel congestion over all testbed nodes and channels.

congestion over all available channels measured at all network nodes is about 9% for 10 interferers and about 29% for 50 interferers. This shows that high interference levels in the network environment can be created and controlled with the number of interferers and the utilized traffic pattern. The short confidence intervals show, that we are able to control the interference level quite well with the number of interferers and the traffic pattern.

Set of available channels

As next, we investigate the impact of the blacklisting mechanism on the set of available channels. The results for the number of available channels for all scenarios is depicted in Figure 8.4.3. For all scenarios, the number of available channels decreases with the growing amount of interferers. This is as expected, since with the higher level of external interference, the mean channel congestion rises and thus more channels are blacklisted. Since DGA does not blacklist any channels, the number of available channels for DGA is 10 for all scenarios.

It is interesting to note, that the number of available channels for 1CBR and 6CBR is similar. With both traffic patterns, the interferer generate traffic with a constant bit rate. Thus, the measuring duration of 1 second per channel is long enough to detect all external interferers. The number of the available channels for BURST is higher. Some interferers are not detected, since the delay between subsequent packet bursts may be longer than the duration of measuring the channel congestion for a single channel. This explains, why the results for EXR for the BURST traffic pattern scenarios are similar to the ones for 6CBR, even though in the latter case much more traffic is generated by the interferers.

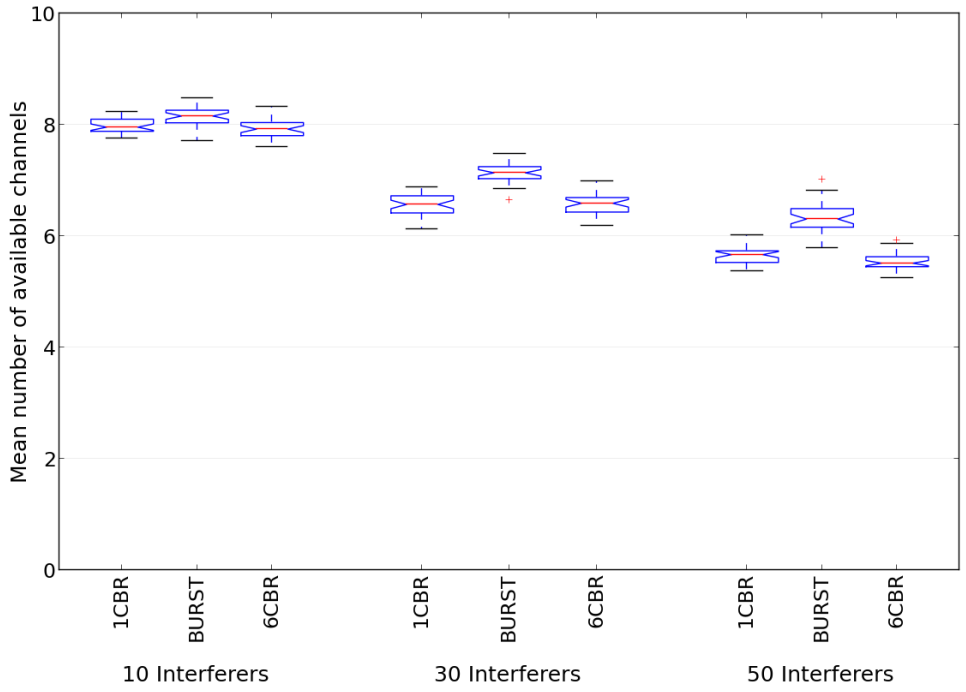


Figure 8.4.3.: Mean amount of available channels after the channel congestion measurement procedure for EICA.

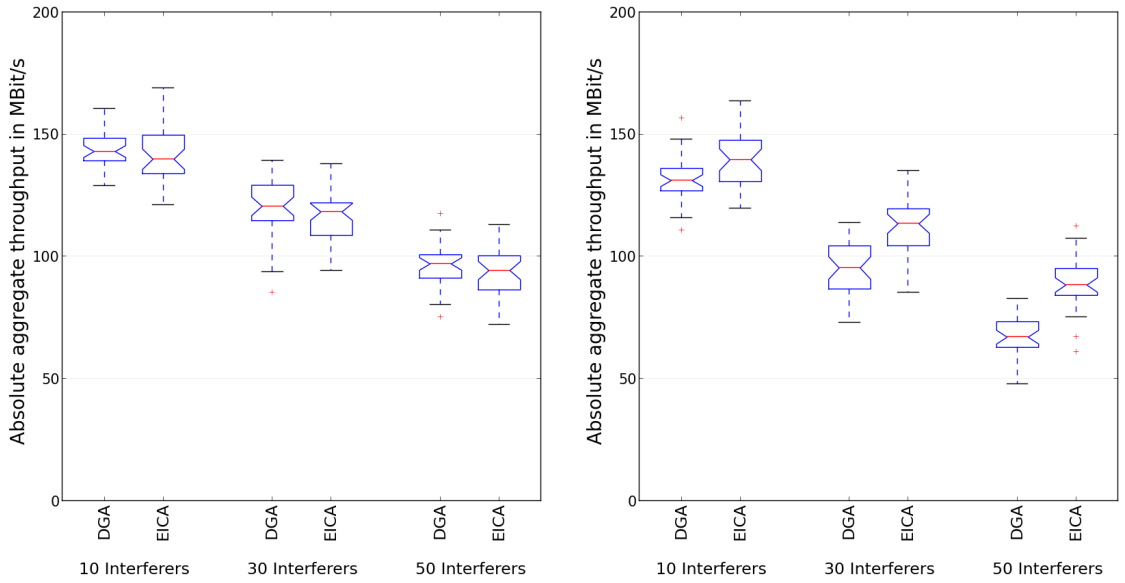
Impact of reduced channel set

The results for EXR as depicted in Figure 8.4.1 have shown, that the blacklisting mechanism of channels with a high load is efficient to calculate a channel assignment that is robust against external interference. However, with the reduction of the amount of available channels, the risk increases, that the network internal interference rises. With more inner-network transmissions interfering with each other, the network performance would also decrease. In order to investigate this issue, we compare the network capacity when the interferers are active to when they are not. Since the set of available channels has been reduced most with 6CBR, the results for this scenario are shown in Figure 8.4.4.

We can observe, that in the case when the interferers are not active, DGA achieves a slightly higher network capacity than EICA. While for 50 interferers the network capacity for DGA is about 94 MBit/s, EICA achieves only 90 MBit/s. Thus, the reduced channel set results in a higher degree of network internal interference. However, considering the case when the interferers are active, we can observe, as previously discussed, that the network capacity with DGA is significantly reduced. With EICA, the network capacity is reduced only by 6% compared to the results, when the interferers are not active.

8.4.4 Discussion

The performance evaluation has shown that the detection of external radio activity can be achieved with spectrum sensing based on the carrier sensing statistics of the WNICs. The resulting external interference-aware channel assignment algorithm EICA has outperformed



(a) Aggregate throughput when the interferers are not active.

(b) Aggregate throughput when the interferers are active.

Figure 8.4.4.: Results for the aggregate throughput when the interferers are active and when they are not for 6CBR. In (a), the aggregate throughput over all links is shown when the interferers are not active, whereas in (b), the aggregate throughput over all links is shown when the interferers are active.

DGA in all 9 scenarios of the performance evaluation in a large-scale multi-radio testbed. EICA is robust even in scenarios with high level of interference, the network capacity decreases only by 6% or less when the interfering nodes are active. Thus EICA is capable to detect channels with a high load and to avoid these channels for the channel assignment.

The results for DGA are as expected since the algorithm does not consider external interference in the channel assignment. This has shown that external interference can significantly decrease the network capacity if it is not considered. With DGA, the network capacity has been reduced up to 30% in scenarios with high levels of interference.

CHAPTER 9

Conclusion

For this dissertation, models and algorithms have been designed, implemented, and validated to enable wireless multi-hop networks to become interference-aware. Initial experiments have shown, that interference is extremely harmful for the performance of wireless multi-hop mesh networks. This includes interference resulting from simultaneous wireless transmissions in the network as well as interference caused by co-located external networks that are not under the control of the network operator. The approach presented in this dissertation has been experimentally validated in a large-scale wireless multi-hop multi-radio testbed. The overall solution significantly increases the network performance by assigning non-overlapping channels for potentially interfering links and by considering the activity of external interferers.

9.1 Results

The overall approach for interference-aware wireless multi-hop mesh networks has been validated in a large-scale multi-radio testbed. The key results are summarized in the following list:

- A scalable measurement-based interference model has been developed and implemented. An experimental evaluation showed that the approach allows efficient and accurate identification of interference relationships between network nodes in large-scale network deployments.

- A software-based spectrum analysis solution has been developed, implemented, and validated that enables the detection of external wireless devices and networks.

- The *external interference-aware channel assignment* (EICA) algorithm has been designed, implemented, and validated in a large-scale wireless multi-hop testbed. It was shown that EICA outperforms other approaches in scenarios with high levels of interference.

- A methodology for performance evaluation with domain-specific performance metrics and scenarios has been developed and proved useful to compare the performance of different channel assignment algorithms.
- A framework for the development process of channel assignment algorithms has been created to allow rapid prototyping of implementation candidates in testbed environments.

9.2 Open issues

The performance evaluation of the *external interference-aware channel assignment* (EICA) showed that the algorithm is capable to reduce significantly the negative impact of external interference on the network performance. The scenarios of the evaluation modeled interference patterns based on CBR traffic and replayed traffic traces. The traffic patterns only comprise a small percentage of what to expect in other environments - different devices such as microwaves, game controllers, and Bluetooth devices might pose additional challenges and might be harder to detect. Thus, further performance evaluations in different scenarios will deliver interesting insights in the wider applicability of the developed approach.

Lastly, the adaptivity of the developed solution in changing network topologies has not been tested yet. These changes include moving network nodes and external devices, as well as dynamic radio activity patterns of external interferers over time. Additionally, long term studies, in which the spectrum sensing phase is repeated periodically or event-based, can be valuable to increase the adaptivity of the developed channel assignment solution.

Bibliography

[1] K. Jain, J. Padhye, V. N. Padmanabhan, and L. Qiu, "Impact of interference on multi-hop wireless network performance," in *Proceedings of the 9th annual international conference on Mobile computing and networking*, ser. MobiCom '03. New York, NY, USA: ACM, 2003, pp. 66–80. [Online]. Available: http://doi.acm.org/10.1145/938985.938993

[2] A. C. V. Gummalla and J. O. Limb, "Wireless medium access control protocols," *Communications Surveys & Tutorials, IEEE*, vol. 3, no. 2, pp. 2–15, 2000.

[3] IEEE Std 802.11-2007, "IEEE standard for information technology — Telecommunications and information exchange between systems — Local and metropolitan area networks-specific requirements — Part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications," LAN/MAN Standards Committee, New York, NY, USA, pp. C1–1184, June 2007. [Online]. Available: http://dx.doi.org/10.1109/IEEESTD.2007.373646

[4] H. Horst and D. Miller, *The cell phone: An anthropology of communication.* Berg publishers, 2006.

[5] J. Haartsen, W. Allen, J. Inouye, O. J. Joeressen, and M. Naghshineh, "Bluetooth: Vision, goals and architecture," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 2, no. 4, p. pp., 1998.

[6] Cisco, "Cisco visual networking index: Global mobile data traffic forecast update, 2012–2017," *Cisco white paper*, 2013. [Online]. Available: http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-520862.pdf

[7] H. Sundmaeker, P. Guillemin, P. Friess, and S. Woelfflé, Eds., *Vision and challenges for realising the Internet of Things.* Cluster of European Research Projects on the Internet of Things, European Commision, 2010.

[8] F. Mattern and C. Floerkemeier, "From the internet of computers to the internet of things," *Lecture Notes in Computer Science - From Active Data Management to Event-Based Systems and More*, vol. 6462, pp. 242–259, 2010.

[9] M. C. V. Ian Fuat Akyildiz, *Wireless Sensor Networks*. John Wiley & Sons Inc, August 2010.

[10] T. S. Rappaport, *Wireless communications, principles and practice*, ser. Communications Engineering and Emerging Technologies. Upper Saddle River, New Jersey, USA: Prentice-Hall, 1996, iSBN: 0-13-375536-3.

[11] V. MacDonald, "The cellular concept," *Bell System Technical Journal*, vol. 58, no. 1, pp. 15–41, 1979.

[12] I. F. Akyildiz, X. Wang, and W. Wang, "Wireless mesh networks: a survey," *Computer Networks*, vol. 47, no. 4, pp. 445–487, March 2005. [Online]. Available: http://www.sciencedirect.com/science/article/B6VRG-4F53V5H-2/2/9fa1587e47665f1fb3f7fb461461dd6b

[13] I. F. Akyildiz and X. Wang, *Wireless Mesh Networks*. Wiley & Sons, March 2009.

[14] K. Fleming, M. Picozzi, C. Milkereit, F. Kühnlenz, B. Lichtblau, J. Fischer, C. Zulfikar, and O. Özel, "The self-organizing seismic early warning information network (SOSEWIN)," *Seismological Research Letters*, vol. 80, no. 5, pp. 755–771, 2009. [Online]. Available: http://srl.geoscienceworld.org/content/80/5/755

[15] D. Raychaudhuri, I. Seskar, M. Ott, S. Ganu, K. Ramachandran, H. Kremo, R. Siracusa, H. Liu, and M. Singh, "Overview of the ORBIT Radio Grid Testbed for Evaluation of Next-Generation Wireless Network Protocols," in *Proc. IEEE Wireless Communications and Networking Conference*, vol. 3, 2005, pp. 1664–1669.

[16] IEEE Std 802.11n-2009, "IEEE Standard for Information technology–Telecommunications and information exchange between systems–Local and metropolitan area networks–Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 5: Enhancements for Higher Throughput," New York, NY, USA, 2009.

[17] D. Gesbert, M. Shafi, D.-s. Shiu, P. J. Smith, and A. Naguib, "From theory to practice: an overview of mimo space-time coded wireless systems," *Selected Areas in Communications, IEEE Journal on*, vol. 21, no. 3, pp. 281–302, 2003.

[18] R. v. Nee and R. Prasad, *OFDM for wireless multimedia communications*. Artech House, Inc., 2000.

[19] IEEE Std 802.11s-2011, "IEEE Standard for Information Technology–Telecommunications and information exchange between systems–Local and metropolitan area networks–Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications Amendment 10: Mesh Networking," New York, NY, USA, 2011.

[20] C. E. Perkins and E. M. Royer, "Ad hoc on-demand distance vector routing," in *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications*, New Orleans, LA, February 1999, Routing, pp. 90–100, `http://moment.cs.ucsb.edu/AODV/aodv.html`.

[21] O. L. P. Child, "One Laptop Per Child Homepage," http://www.laptop.org/, last visit: 2013.

[22] IEEE Std 802.16-2009, "IEEE Standard for Local and metropolitan area networks — Part 16: Air Interface for Broadband Wireless Access Systems," LAN/MAN Standards Committee, New York, NY, USA, pp. C1–1184, 2009.

[23] computerworld.com, "Cisco quits WiMax radio business," http://www.computerworld.com/s/article/9167098, last visit: 2013.

[24] "Homepage of BelAir Networks," http://www.belairnetworks.com/, last visit: 2013.

[25] "Homepage of Meraki," http://meraki.com/, last visit: 2013.

[26] "Homepage of Tropos," http://www.tropos.com/, last visit: 2013.

[27] Libelium, "Homepage of libelium," http://www.libelium.com, last visit: 2013.

[28] "Freifunk," http://start.freifunk.net/, last visit: 2013.

[29] J. Bicket, D. Aguayo, S. Biswas, and R. Morris, "Architecture and evaluation of an unplanned 802.11 b mesh network," in *Proceedings of the 11th annual international conference on Mobile computing and networking*. ACM, 2005, pp. 31–42.

[30] R. Sombrutzki, A. Zubow, M. Kurth, and J. Redlich, "Self-organization in community mesh networks the berlin roofnet," in *Operator-Assisted (Wireless Mesh) Community Networks, 2006 1st Workshop on*. IEEE, 2006, pp. 1–11.

[31] M. Portmann and A. A. Pirzada, "Wireless mesh networks for public safety and crisis management applications," *IEEE Internet Computing*, vol. 12, no. 1, pp. 18–25, Jan. 2008. [Online]. Available: http://dx.doi.org/10.1109/MIC.2008.25

[32] J. Fischer, J.-P. Redlich, B. Scheuermann, J. Schiller, M. Günes, K. Nagel, P. Wagner, M. Scheidgen, A. Zubow, I. Eveslage, R. Sombrutzki, and F. Juraschek, "From Earthquake Detection to Traffic Surveillance–About Information and Communication Infrastructures for Smart Cities," in *System Analysis and Modeling: Theory and Practice*. Springer, 2013, pp. 121–141.

[33] M. Scheidgen, A. Zubow, and R. Sombrutzki, "Clickwatch – an experimentation framework for communication network test-beds," in *IEEE Wireless Communications and Networking Conference (WCNC)*. Paris: IEEE, 2012.

[34] M. Güneş, F. Juraschek, B. Blywis, Q. Mushtaq, and J. Schiller, "A Testbed for Next Generation Wireless Networks Research," *Special Issue PIK on Mobile Ad-hoc Networks*, vol. IV, pp. 208–212, October-December 2009. [Online]. Available: http://www.reference-global.com/doi/abs/10.1515/piko.2009.0040

[35] P. Gupta and P. R. Kumar, "The capacity of wireless networks," *IEEE Transactions on Information Theory*, vol. 46, no. 2, pp. 388–404, 2000.

[36] A. P. Subramanian, H. Gupta, S. R. 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.

[37] S. Rayanchu, A. Patro, and S. Banerjee, "Airshark: detecting non-WiFi RF devices using commodity WiFi hardware," in *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference*. New York, NY, USA: ACM, 2011, pp. 137–154. [Online]. Available: http://doi.acm.org/10.1145/2068816.2068830

[38] D. Gokhale, S. Sen, K. Chebrolu, and B. Raman, "On the feasibility of the link abstraction in (rural) mesh networks," in *The 27th Conference on Computer Communications (INFOCOM)*. IEEE, 2008, pp. 61–65.

[39] B. N. Clark, C. J. Colbourn, and D. S. Johnson, "Unit disk graphs," *Discrete Mathematics*, vol. 86, no. 1-3, pp. 165–177, December 1990. [Online]. Available: http://dx.doi.org/10.1016/0012-365X(90)90358-O

[40] S. Sridhar, J. Guo, and S. Jha, "Channel Assignment in Multi-Radio Wireless Mesh Networks : A Graph-theoretic approach," in *Communication Systems and Networks and Workshops (COMSNETS)*, 2009.

[41] W. Si, S. Selvakennedy, and A. Y. Zomaya, "An overview of channel assignment methods for multi-radio multi-channel wireless mesh networks," *Journal of Parallel and Distributed Computing*, vol. 70, pp. 505–524, May 2010.

[42] F. Juraschek, M. Güneş, and B. Blywis, "External Interference-Aware Distributed Channel Assignment in Wireless Mesh Networks," in *The Fifth International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM)*, Lissabon, 2011.

[43] P. Pawelczak, K. Nolan, L. Doyle, S. W. Oh, and D. Cabric, "Cognitive radio ten years of experimentation and development," *Communications Surveys Tutorials, IEEE*, vol. 11, pp. 90–100, 2009.

[44] ZigBee Alliance, "ZigBee Core Specification," http://www.zigbee.org/Specifications.aspx, last checked: 2013.

[45] Bluetooth-SIG, *Specification of the Bluetooth System - Core*, 1999.

[46] J. Padhye, S. Agarwal, V. N. Padmanabhan, L. Qiu, A. Rao, and B. Zill, "Estimation of link interference in static multi-hop wireless networks," in *IMC '05: Proceedings of the 5th ACM SIGCOMM conference on Internet Measurement*. USA: USENIX Association, 2005.

[47] M. Kodialam and T. Nandagopal, "Characterizing achievable rates in multi-hop wireless networks: the joint routing and scheduling problem," in *Proceedings of the 9th annual international conference on Mobile computing and networking*. New York, NY, USA: ACM, 2003, pp. 42–54. [Online]. Available: http://doi.acm.org/10.1145/938985.938991

[48] B.-J. Ko, V. Misra, J. Padhye, and D. Rubenstein, "Distributed channel assignment in multi-radio 802.11 mesh networks," in *Wireless Communications and Networking Conference (WCNC)*, 2007, pp. 3978–3983.

[49] D. Kotz, C. Newport, R. S. Gray, J. Liu, Y. Yuan, and C. Elliott, "Experimental evaluation of wireless simulation assumptions," in *Proceedings of the 7th ACM international symposium on Modeling, analysis and simulation of wireless and mobile systems*. New York, NY, USA: ACM, 2004, pp. 78–82. [Online]. Available: http://doi.acm.org/10.1145/1023663.1023679

[50] F. Juraschek, M. Güneş, and B. Blywis, "Measurement-Based Interference Modeling Using Channel Occupancy in Wireless Mesh Networks," in *Inproceedings of The 13th IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, San Francisco, 2012.

[51] D. Niculescu, "Interference map for 802.11 networks," in *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, ser. IMC '07. New York, NY, USA: ACM, 2007, pp. 339–350. [Online]. Available: http://doi.acm.org/10.1145/1298306.1298355

[52] I. Katzela and M. Naghshineh, "Channel assignment schemes for cellular mobile telecommunication systems," *IEEE Personal Communications*, vol. 3, pp. 10–31, 1996.

[53] K. N. Ramachandran, E. M. Belding, K. C. Almeroth, and M. M. Buddhikot, "Interference-Aware Channel Assignment in Multi-Radio Wireless Mesh Networks," in *25th IEEE International Conference on Computer Communications (INFOCOM)*, 2006, pp. 1–12. [Online]. Available: http://dx.doi.org/10.1109/INFOCOM.2006.177

[54] A. Raniwala, K. Gopalan, and T. Chiueh, "Centralized channel assignment and routing algorithms for multi-channel wireless mesh networks," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 8, no. 2, pp. 50–65, 2004.

[55] M. K. Marina and S. R. Das, "A topology control approach for utilizing multiple channels in multi-radio wireless mesh networks," in *In Broadnets*, 2005, pp. 381–390.

[56] A. Raniwala and T. Chiueh, "Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network," in *Proc. IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies INFOCOM 2005*, T.-c. Chiueh, Ed., vol. 3, 2005, pp. 2223 – 2234.

[57] J. Crichigno, M.-Y. Wu, and W. Shu, "Protocols and architectures for channel assignment in wireless mesh networks," *Ad Hoc Networks*, vol. 6, no. 7, pp. 1051 – 1077, 2008. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1570870507001564

[58] A. Prodran and V. Mirchandani, *Guide to Wireless Mesh Networks.* Springer, 2009, ch. Channel Assignment Techniques for 802.11-Based Multiradio Wireless Mesh Networks.

[59] C. C. Pradeep Kyasanur and N. H. Vaidya, "Net-x: System extensions for supporting multiple channels, multiple interfaces, and other interface capabilities," University of Illinois at Urbana-Champaign, Tech. Rep., August 2006.

[60] R. Chandra, V. Bahl, and P. Bahl, "Multinet: Connecting to multiple ieee 802.11 networks using a single wireless card," in *IEEE INFOCOM*, 2004.

[61] Y. Ding and L. Xiao, "Channel allocation in multi-channel wireless mesh networks," *Computer Communications*, vol. 34, no. 7, pp. 803–815, 2011.

[62] B.-J. Ko, V. Misra, J. Padhye, and D. Rubenstein, "Distributed channel assignment in multi-radio 802.11 mesh networks," Columbia University, Tech. Rep., June 2006.

[63] A. M. Frieze and M. Jerrum, "Improved approximation algorithms for max k-cut and max bisection," in *Proceedings of the 4th International IPCO Conference on Integer Programming and Combinatorial Optimization.* London, UK, UK: Springer-Verlag, 1995, pp. 1–13. [Online]. Available: http://dl.acm.org/citation.cfm?id=645586.659459

[64] S. Kim, J. Cha, and J. Ma, "Interference-aware channel assignments with seamless multi-channel monitoring in wireless mesh networks," in *Proceedings of the 2009 IEEE international conference on Communications*, ser. ICC'09. Piscataway, NJ, USA: IEEE Press, 2009.

[65] P. Kyasanur, J. So, C. Chereddi, and N. H. Vaidya, "Multichannel mesh networks: challenges and protocols," *Wireless Communications, IEEE [see also IEEE Personal Communications]*, vol. 13, no. 2, pp. 30–36, 2006. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1632478

[66] P. Kyasanur and N. H. Vaidya, "Routing and link-layer protocols for multi-channel multi-interface ad hoc wireless networks," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 10, no. 1, pp. 31–43, 2006.

[67] W. Kim, A. J. Kassler, M. Di Felice, M. Gerla, and L. Bononi, "Urban-x: a self-organizing cognitive wireless mesh network for dense city environments," in *Proceedings of the 9th IFIP TC 6 international conference on Wired/wireless internet communications*, ser. WWIC'11. Berlin, Heidelberg: Springer-Verlag, 2011, pp. 398–409. [Online]. Available: http://dl.acm.org/citation.cfm?id=2023094.2023134

[68] W. Kim, A. Kassler, M. Di Felice, and M. Gerla, "Urban-x: towards distributed channel assignment in cognitive multi-radio mesh networks," in *Proceedings of the IFIP Wireless Days, Venice, Italy*, 2010.

[69] M. Shin, S. Lee, and Y. ah Kim, "Distributed channel assignment for multi-radio wireless networks," *IEEE International Conference on Mobile Adhoc and Sensor Systems Conference*, pp. 417–426, 2006.

[70] S. A. Makram, M. Günes, M. Wenig, and A. Zimmermann, "Adaptive Channel Assignment to Support QoS and Load Balancing for Wireless Mesh Networks," RWTH Aachen, Tech. Rep. AIB-2007-16, 2007.

[71] S. Das, S. Das, H. Pucha, D. Koutsonikolas, Y. Hu, and D. Peroulis, "Dmesh: Incorporating practical directional antennas in multichannel wireless mesh networks," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 11, pp. 2028–2039, 2006.

[72] R. Draves, J. Padhye, and B. Zill, "Routing in multi-radio, multi-hop wireless mesh networks," in *MobiCom '04: Proceedings of the 10th annual international conference on Mobile computing and networking.* New York, NY, USA: ACM, 2004, pp. 114–128.

[73] R. Maheshwari, J. Cao, and S. Das, "Physical interference modeling for transmission scheduling on commodity wifi hardware," in *IEEE INFOCOM 2009*, 2009.

[74] MadWifi project, "MadWifi project homepage," Online, last checked: 2013.

[75] A. Sharma and E. M. Belding, "Freemac: framework for multi-channel mac development on 802.11 hardware," in *Proceedings of the ACM workshop on Programmable routers for extensible services of tomorrow*, ser. PRESTO '08. New York, NY, USA: ACM, 2008, pp. 69–74. [Online]. Available: http://doi.acm.org/10.1145/1397718.1397734

[76] A. Sharma, M. Tiwari, and H. Zheng, "Madmac: Building a reconfiguration radio testbed using commodity 802.11 hardware," in *Networking Technologies for Software Defined Radio Networks, 2006. SDR'06.1 st IEEE Workshop on.* IEEE, 2006, pp. 78–83.

[77] Wireless Networking Group at UIUC, "Net-x channel assignment framework," http://www.crhc.illinois.edu/wireless/netx.html, Last visit: 2010.

[78] V. Raman and N. H. Vaidya, "Adjacent channel interference reduction in multi-channel wireless networks using intelligent channel allocation," Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, Tech. Rep., August 2009.

[79] M. C. Castro, P. Dely, A. J. Kassler, and N. H. Vaidya, "Qos-aware channel scheduling for multi-radio/multi-channel wireless mesh networks," in *WINTECH '09: Proceedings of the 4th ACM international workshop on Experimental evaluation and characterization.* New York, NY, USA: ACM, 2009, pp. 11–18.

[80] M. Baar, H. Will, B. Blywis, A. Liers, G. Wittenburg, and J. Schiller, "The ScatterWeb MSB-A2 Platform for Wireless Sensor Networks," Freie Universität Berlin, Tech. Rep., 2008.

[81] OPNEX consortium, "OPNEX - Optimization driven Multi-Hop Network Design and Experimentation," http://www.opnex.eu/, last visit: 2013.

[82] WISEBED consortium, "WISEBED - Wireless Sensor Network Testbeds," http://wisebed.eu/, last visit: 2013.

[83] G-Lab consortium, "G-Lab," http://www.german-lab.de/, last visit: 2013.

[84] "Ralink technology, Ralink RT2501U chipset product specification." [Online]. Available: http://www.ralinktech.com/

[85] Linux wireless, "ath5k driver homepage," http://linuxwireless.org/en/users/Drivers/ath5k, last visit: 2013.

[86] E. T. S. Institute, "European telecommunications standards institute homepage," http://www.etsi.org, Last visit: 2013.

[87] F. C. Commission, "Federal communications commission homepage," http://www.fcc.gov/, Last visit: 2013.

[88] Telecom Engineering Center, "Telec," http://www.telec.or.jp/, last visit: 2013.

[89] M. Güneş, B. Blywis, and F. Juraschek, "Concept and Design of the Hybrid Distributed Embedded Systems Testbed," Freie Universität Berlin, Tech. Rep. TR-B-08-10, August 2008. [Online]. Available: ftp://ftp.inf.fu-berlin.de/pub/reports/tr-b-08-10.pdf

[90] M. Güneş, B. Blywis, F. Juraschek, and P. Schmidt, "Practical Issues of Implementing a Hybrid Multi-NIC Wireless Mesh-Network," Freie Universität Berlin, Tech. Rep. TR-B-08-11, August 2008. [Online]. Available: ftp://ftp.inf.fu-berlin.de/pub/reports/tr-b-08-11.pdf

[91] B. Blywis, M. Güneş, F. Juraschek, O. Hahm, and N. Schmittberger, "Properties and Topology of the DES-Testbed (2nd Extended Revision)," Freie

Universität Berlin, Tech. Rep. TR-B-11-04, July 2011. [Online]. Available: http://edocs.fu-berlin.de/docs/receive/FUDOCS_document_000000010697

[92] G. W. Cobb, *Introduction to Design and Analysis of Experiments.* Key College, 2002. [Online]. Available: http://www.keycollege.com/catalog/titles/intro_to_design_analysis.html

[93] D. C. Montgomery, *Design and Analysis of Experiments*, 7th ed. Wiley, 2009.

[94] T. Rakotoarivelo, M. Ott, G. Jourjon, and I. Seskar, "OMF: a control and management framework for networking testbeds," *SIGOPS Oper. Syst. Rev.*, vol. 43, pp. 54–59, January 2010. [Online]. Available: http://doi.acm.org/10.1145/1713254.1713267

[95] M. Güneş, F. Juraschek, and B. Blywis, "An Experiment Description Language for Wireless Network Research," *Journal of Internet Technology (JIT), Special Issue for Mobile Internet*, July 2010.

[96] M. Güneş, B. Blywis, F. Juraschek, and O. Watteroth, "Experimentation Made Easy," in *Inproceedings of the First International Conference on Ad Hoc Networks*, Ontario, Canada, September 2009.

[97] M. Güneş, F. Juraschek, B. Blywis, and O. Watteroth, "DES-CRIPT - A Domain Specific Language for Network Experiment Descriptions," in *Inproceedings of the International Conference on Next Generation Wireless Systems (NGWS)*, Melbourne, Australia, October 2009.

[98] JavaView, "JavaView - Interactive 3D Geometry and Visualization," http://www.javaview.de/, last visit: 2013.

[99] R project, "The R Project for Statistical Computing," http://www.r-project.org/, last visit: 2013. [Online]. Available: http://www.r-project.org/

[100] D. S. J. De Couto, D. Aguayo, J. Bicket, and R. Morris, "A high-throughput path metric for multi-hop wireless routing," in *MobiCom '03: Proceedings of the 9th annual international conference on Mobile computing and networking.* New York, NY, USA: ACM, 2003, pp. 134–146.

[101] P. Fuxjager, D. Valerio, and F. Ricciato, "The myth of non-overlapping channels: interference measurements in IEEE 802.11," in *Proc. Fourth Annual Conference on Wireless on Demand Network Systems and Services WONS '07*, 2007, pp. 1–8.

[102] P. Dely, M. Castro, S. Soukhakian, A. Moldsvor, and A. Kassler, "Practical considerations for channel assignment in wireless mesh networks," in *GLOBECOM Workshops (GC Wkshps), 2010 IEEE.* IEEE, 2010, pp. 763–767.

[103] V. Angelakis, S. Papadakis, V. Siris, and A. Traganitis, "Adjacent channel interference in 802.11a is harmful. testbed validation of a simple quantification model." *Communications Magazine, IEEE*, vol. 49, no. 3, pp. 160–166, March 2011.

[104] J. Robinson, K. Papagiannaki, C. Diota, X. Guo, and L. Krishnamurthy, "Experimenting with a multi-radio mesh networking testbed," in *In Proceedings of the First Workshop on Wireless Network Measurements (WiNMee 2005)*, 2005.

[105] F. Juraschek, S. Seif, M. Güneş, and B. Blywis, "A Benchmark for Channel Assignment Algorithms in Wireless Testbeds," in *Inproceedings of The 8th International Workshop on Wireless Network Measurements (WiNMee)*, Paderborn, May 2012.

[106] J. E. Smith, "Characterizing computer performance with a single number," *Commun. ACM*, vol. 31, pp. 1202–1206, October 1988.

[107] S. Bouckaert, J. V. V. Gerwen, I. Moerman, S. C. Phillips, and J. Wilander, "Benchmarking computers and computer networks," EU FIRE White Paper, 2010.

[108] S. Makram and M. Günes, "Distributed Channel Assignment for Multi-Radio Wireless Mesh-Networks," in *Proceedings of IEEE Symposium on Computers and Communications (ISCC'08)*. Marrakech, Morocco: IEEE Computer Society Press, July 2008.

[109] F. Juraschek, M. Güneş, M. Philipp, and B. Blywis, "State-of-the-Art of Distributed Channel Assignment," Freie Universität Berlin, FB Mathematik und Informatik, Tech. Rep. TR-B-11-01, Jan 2011. [Online]. Available: http://edocs.fu-berlin.de/docs/receive/FUDOCS_document_000000009172

[110] G. Bianchi, "IEEE 802.11-saturation throughput analysis," *Communications Letters, IEEE*, vol. 2, no. 12, pp. 318–320, 1998. [Online]. Available: http://dx.doi.org/10.1109/4234.736171

[111] P. Subramanian, M. M. Buddhikot, and S. Miller, "Interference aware routing in multi-radio wireless mesh networks," in *In Proc. of IEEE Workshop on Wireless Mesh Networks (WiMesh*, 2006, pp. 55–63.

[112] W. Jiang, S. Liu, Y. Zhu, and Z. Zhang, "Optimizing routing metrics for large-scale multi-radio mesh networks," in *International Conference on Wireless Communications, Networking and Mobile Computing, 2007 (WiCom)*, 2007, pp. 1550–1553. [Online]. Available: http://dx.doi.org/10.1109/WICOM.2007.390

[113] M. Alicherry, R. Bhatia, and L. E. Li, "Joint channel assignment and routing for throughput optimization in multi-radio wireless mesh networks," in *MobiCom '05: Proceedings of the 11th annual international conference on Mobile computing and networking*. New York, NY, USA: ACM, 2005, pp. 58–72.

[114] X. Lin and S. Rasool, "A distributed joint channel-assignment,scheduling and routing algorithm for multi-channel ad hoc wireless networks," in *26th IEEE International Conference on Computer Communications (INFOCOM)*, 2007.

[115] 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*, pp. 1972–1983, 2006.

[116] iperf community, "iperf," http://iperf.sourceforge.net/, last visit: 2013.

[117] nuttcp development team, "nuttcp," http://www.nuttcp.net, last visit: 2013.

[118] M. Muuss and T. Slattery, "ttcp," http://ftp.arl.mil/ mike/ttcp.html, last visit: 2013.

[119] F. Juraschek, M. Güneş, M. Philipp, and B. Blywis, "On the Feasibility of Distributed Link-Based Channel Assignment in Wireless Mesh Networks," in *Proceedings of the 9th ACM International Symposium on Mobility Management and Wireless Access (MobiWac)*, November 2011.

[120] S. Waharte, B. Ishibashi, R. Boutaba, and D. Meddour, "Interference-Aware Routing Metric for Improved Load Balancing in Wireless Mesh Networks," in *IEEE International Conference onCommunications, 2008 (ICC '08)*, 2008, pp. 2979–2983. [Online]. Available: http://dx.doi.org/10.1109/ICC.2008.561

[121] K. Jamieson, B. Hull, A. Miu, and H. Balakrishnan, "Understanding the real-world performance of carrier sense," in *Proceedings of the 2005 ACM SIGCOMM workshop on Experimental approaches to wireless network design and analysis*, ser. E-WIND '05. New York, NY, USA: ACM, 2005, pp. 52–57. [Online]. Available: http://dx.doi.org/10.1145/1080148.1080160

[122] C. Reis, R. Mahajan, M. Rodrig, D. Wetherall, and J. Zahorjan, "Measurement-based models of delivery and interference in static wireless networks," *SIGCOMM Comput. Commun. Rev.*, vol. 36, no. 4, pp. 51–62, 2006.

[123] A. Kashyap, "A measurement-based approach to modeling link capacity in 802.11-based wireless networks," in *ACM MOBICOM 2007*. ACM Press, 2007, pp. 242–253.

[124] D. Son, B. Krishnamachari, and J. Heidemann, "Experimental study of concurrent transmission in wireless sensor networks," in *Proceedings of the 4th international conference on Embedded networked sensor systems*, ser. SenSys '06. New York, NY, USA: ACM, 2006, pp. 237–250. [Online]. Available: http://doi.acm.org/10.1145/1182807.1182831

[125] L. Qiu, Y. Zhang, F. Wang, M. K. Han, and R. Mahajan, "A general model of wireless interference," in *Proceedings of the 13th annual ACM international conference on Mobile computing and networking*, ser. MobiCom '07. New York, NY, USA: ACM, 2007, pp. 171–182. [Online]. Available: http://doi.acm.org/10.1145/1287853.1287874

[126] *The Python Twisted Documentation*, Last visit: 2013. [Online]. Available: http://twistedmatrix.com/projects/core/documentation/howto/book.pdf

[127] F. Juraschek, M. Güneş, M. Philipp, B. Blywis, and O. Hahm, "DES-Chan: A Framework for Distributed Channel Assignment in Wireless Mesh Networks," in *Proceedings of the Australasian Telecommunication Networks And Applications Conference (ATNAC 2011)*, Melbourne, November 2011.

[128] F. Juraschek, M. Güneş, and B. Blywis, "A Framework for External Interference-Aware Distributed Channel Assignment," *International Journal of Wireless Networks and Broadband Technologies (IJWNBT)*, vol. 1, no. 4, pp. 40–54, 2012.

[129] R. Joost and S. Robinson, "pythonwifi," http://pythonwifi.wikispot.org/, Last visit: 2013.

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

[131] G. Project, "Graphviz project," http://www.graphviz.org/, last visit: 2013.

[132] M. Philipp, "A Framework for Distributed Channel Assignment in Wireless Mesh Networks," Freie Universität Berlin, Masterthesis, May 2010.

[133] B.-J. Ko and D. Rubenstein, "A distributed, self-stabilizing protocol for placement of replicated resources in emerging networks," *IEEE/ACM Transactions on Networking*, vol. 13, no. 3, June 2005.

[134] S. Seif, "Implementation and evaluation of interface-based distributed channel assignment in wireless mesh networks," Freie Universität Berlin, Bachelorthesis, Nov 2011.

[135] WiSpy, "WiSpy homepage," http://www.wi-spy.co.uk/, last visit: 2013.

[136] Spectrum XT, "Spectrum XT homepage," http://www.flukenetworks.com/products/airmagnet-spectrum-xt, last visit: 2013.

[137] IEEE Std 802.11k-2008, "IEEE standard for information technology — Telecommunications and information exchange between systems — Local and metropolitan area networks-specific requirements — Part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications Amendment 1: Radio Resource Measurement of Wireless LANs," LAN/MAN Standards Committee, New York, NY, USA, pp. C1–1184, 2008.

[138] S. Mangold and L. Berlemann, "IEEE 802.11k: improving confidence in radio resource measurements," in *Personal, Indoor and Mobile Radio Communications,*

2005. PIMRC 2005. IEEE 16th International Symposium on, vol. 2, 2005, pp. 1009–1013 Vol. 2. [Online]. Available: http://dx.doi.org/10.1109/PIMRC.2005.1651593

[139] P. Dely, A. Kassler, and D. Sivchenko, "Theoretical and experimental analysis of the channel busy fraction in ieee 802.11," in *Proceedings of Future Network & Mobile Summit 2010*. IEEE, 2010, pp. 1–9.

[140] P. A. Acharya, A. Sharma, E. M. Belding, K. C. Almeroth, and K. D. Papagiannaki, "Rate adaptation in congested wireless networks through real-time measurements," *IEEE Transactions on Mobile Computing*, vol. 9, pp. 1535–1550, 2010.

[141] T. Salonidis, M. Garetto, A. Saha, and E. Knightly, "Identifying High Throughput Paths in 802.11 Mesh Networks: a Model-based Approach," in *Network Protocols, 2007. ICNP 2007. IEEE International Conference on*, 2007, pp. 21–30. [Online]. Available: http://dx.doi.org/10.1109/ICNP.2007.4375833

[142] P. A. K. Acharya, A. Sharma, E. M. Belding, K. C. Almeroth, and K. Papagiannaki, "Congestion-aware rate adaptation in wireless networks: A measurement-driven approach," in *5th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON '08)*, 2008, pp. 1–9.

[143] K. Lakshminarayanan, S. Seshan, and P. Steenkiste, "Understanding 802.11 performance in heterogeneous environments," in *Proceedings of the 2nd ACM SIGCOMM workshop on Home networks*, ser. HomeNets '11. New York, NY, USA: ACM, 2011, pp. 43–48. [Online]. Available: http://doi.acm.org/10.1145/2018567.2018577

[144] D. Giustiniano and S. Mangold, "Caesar: carrier sense-based ranging in off-the-shelf 802.11 wireless lan," in *Proceedings of the Seventh COnference on emerging Networking EXperiments and Technologies*, ser. CoNEXT '11. New York, NY, USA: ACM, 2011, pp. 10:1–10:12. [Online]. Available: http://doi.acm.org/10.1145/2079296.2079306

List of Publications

[1] F. Juraschek and M. Güneş, "External interference-aware channel assignment," in *The 9th International Wireless Communications & Mobile Computing Conference (IWCMC)*. Cagliari, Italy: IEEE, July 2013.

[2] F. Juraschek, S. Seif, and M. Güneş, "Distributed Channel Assignment in Large-Scale Wireless Mesh Networks: A Performance Analysis," in *IEEE International Conference on Communications (ICC)*, June 2013.

[3] F. Juraschek, "Interference-Aware Wireless Mesh Networks," in *Inproceedings of The 14th IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM) (PhD Forum)*, Madrid, 2013.

[4] J. Fischer, J.-P. Redlich, B. Scheuermann, J. Schiller, M. Günes, K. Nagel, P. Wagner, M. Scheidgen, A. Zubow, I. Eveslage, R. Sombrutzki, and F. Juraschek, "From Earthquake Detection to Traffic Surveillance–About Information and Communication Infrastructures for Smart Cities," in *System Analysis and Modeling: Theory and Practice*. Springer, 2013, pp. 121–141.

[5] F. Juraschek, M. Güneş, and B. Blywis, "Measurement-Based Interference Modeling Using Channel Occupancy in Wireless Mesh Networks," in *Inproceedings of The 13th IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, San Francisco, 2012.

[6] ——, "A Framework for External Interference-Aware Distributed Channel Assignment," *International Journal of Wireless Networks and Broadband Technologies (IJWNBT)*, vol. 1, no. 4, pp. 40–54, 2012.

[7] F. Juraschek, S. Seif, M. Güneş, and B. Blywis, "A Benchmark for Channel Assignment Algorithms in Wireless Testbeds," in *Inproceedings of The 8th International Workshop on Wireless Network Measurements (WiNMee)*, Paderborn, May 2012.

[8] F. Juraschek, A. Zubow, O. Hahm, M. Scheidgen, B. Blywis, R. Sombrutzki, M. Güneş, and J. Fischer, "Towards Smart Berlin - an Experimental Facility for Heterogeneous Smart City Infrastructures," in *Proceedings of 1st Workshop on GlObal Trends in Smart Cities 2012 (goSMART)*, Clearwater, FL, October 2012.

[9] K. Choumasa, S. Keranidis, T. Korakis, I. Koutsopoulos, L. Tassiulas, F. Juraschek, M. Günes, E. Baccelli, P. Misiorek, A. Szwabe, T. Salonidis, and H. Lundgren, "Optimization driven Multi-Hop Network Design and Experimentation: The Approach of the FP7 Project OPNEX," *IEEE Communications Magazine, Radio Communication Series*, vol. 50, no. 6, pp. 122–130, 2012.

[10] A. Szwabe, P. Misiorek, M. Urbanski, F. Juraschek, and M. Güneş, "Multi-path OLSR Performance Analysis in a Large Testbed Environment," in *13th International Conference on Distributed Computing and Networking*, Hong Kong, January 2012.

[11] F. Juraschek, M. Güneş, M. Philipp, and B. Blywis, "Insights from Experimental Research on Distributed Channel Assignment in Wireless Testbeds," *International Journal of Wireless Networks and Broadband Technologies (IJWNBT)*, vol. 1, no. 1, pp. 32–49, 2011.

[12] ——, "On the Feasibility of Distributed Link-Based Channel Assignment in Wireless Mesh Networks," in *Proceedings of the 9th ACM International Symposium on Mobility Management and Wireless Access (MobiWac)*, November 2011.

[13] F. Juraschek, M. Güneş, M. Philipp, B. Blywis, and O. Hahm, "DES-Chan: A Framework for Distributed Channel Assignment in Wireless Mesh Networks," in *Proceedings of the Australasian Telecommunication Networks And Applications Conference (ATNAC 2011)*, Melbourne, November 2011.

[14] F. Juraschek, M. Güneş, and B. Blywis, "External Interference-Aware Distributed Channel Assignment in Wireless Mesh Networks," in *The Fifth International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM)*, Lissabon, 2011.

[15] F. Juraschek, M. Güneş, M. Philipp, and B. Blywis, "State-of-the-Art of Distributed Channel Assignment," Freie Universität Berlin, FB Mathematik und Informatik, Tech. Rep. TR-B-11-01, Jan 2011. [Online]. Available: http://edocs.fu-berlin.de/docs/receive/FUDOCS_document_000000009172

[16] H. Will, F. Juraschek, M. Güneş, and J. Schiller, "Experiences from the MANIAC Challenge," in *Proceedings of the Australasian Telecommunication Networks And Applications Conference (ATNAC 2011)*, Melbourne, November 2011.

[17] O. Hahm, M. Güneş, F. Juraschek, B. Blywis, and N. Schmittberger, "An Experimental Facility for Wireless Multi-Hop Networks in Future Internet scenarios," in *The 2011 IEEE International Conference on Internet of Things (iThings 2011)*, Dalian, China, October 2011.

[18] B. Blywis, M. Güneş, F. Juraschek, O. Hahm, and N. Schmittberger, "A Survey of Flooding, Gossip Routing, and Related Schemes for Wireless Multi-hop Networks,"

Freie Universität Berlin, Tech. Rep. TR-B-11-06, October 2011. [Online]. Available: http://edocs.fu-berlin.de/docs/receive/FUDOCS_document_000000011892

[19] B. Blywis, M. Güneş, F. Juraschek, and O. Hahm, "Challenges and Limits of Flooding and Gossip Routing Based Route Discovery Schemes," in *The 36th IEEE Conference on Local Computer Networks (LCN)*, Bonn, 2011.

[20] B. Blywis, M. Güneş, F. Juraschek, O. Hahm, and N. Schmittberger, "Properties and Topology of the DES-Testbed (2nd Extended Revision)," Freie Universität Berlin, Tech. Rep. TR-B-11-04, July 2011. [Online]. Available: http://edocs.fu-berlin.de/docs/receive/FUDOCS_document_000000010697

[21] B. Blywis, M. Güneş, F. Juraschek, and J. Schiller, "Trends, Advances, and Challenges in Testbed-based Wireless Mesh Network Research," *Mobile Networks and Applications*, vol. 15, pp. 315–329, 2010, 10.1007/s11036-010-0227-9. [Online]. Available: http://dx.doi.org/10.1007/s11036-010-0227-9

[22] B. Blywis, M. Güneş, F. Juraschek, and S. Gliech, "A Localization Framework for wireless mesh networks - Anchor-Free Distributed Localization in the DES-Testbed," in *Indoor Positioning and Indoor Navigation (IPIN), 2010 International Conference on*, September 2010, pp. 1–10.

[23] M. Güneş, F. Juraschek, and B. Blywis, "An Experiment Description Language for Wireless Network Research," *Journal of Internet Technology (JIT), Special Issue for Mobile Internet*, July 2010.

[24] B. Blywis, M. Güneş, F. Juraschek, and S. Gliech, "A Localization Framework for Wireless Mesh Networks," in *International Conference on Indoor Positioning and Indoor Navigation (IPIN) Abstract Volume*. ETH Zürich, September 2010, pp. 201–202.

[25] B. Blywis, M. Güneş, S. Hofmann, and F. Juraschek, "A Study of Adaptive Gossip Routing in Wireless Mesh Networks," in *Ad Hoc Networks*, ser. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, vol. 49. Springer Berlin Heidelberg, 2010, pp. 98–113. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-17994-5_7

[26] B. Blywis, M. Güneş, D. J. H. Gutzmann, and F. Juraschek, "A testbed-based study of uni- and multi-path Dynamic Source Routing in a WMN," in *Wireless Days (WD), 2010 IFIP*, October 2010, pp. 1–5.

[27] M. Güneş, F. Juraschek, B. Blywis, and C. Graff, "MoNoTrac - A Mobility Trace Generator Based On OpenStreetMap Geo-Data," in *Second Workshop on Scenarios for Network Evaluation Studies (SCENES 2010)*, San Francisco, 2010.

[28] B. Blywis, F. J. Mesut Güneş, and S. Hofmann, "Gossip Routing in Wireless Mesh Networks," in *IEEE 21st International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)*, September 2010, pp. 1572–1577.

[29] F. Juraschek, H. Will, M. Güneş, and J. Schiller, "Friendly Clustering - The Winning Strategy of the MANIAC Challenge 2009," in *Inproceedings of the Fourth ACM International Workshop on Wireless Network Testbeds, Experimental Evaluation and Characterization (WiNTECH)*, Beijing, China, September 2009.

[30] F. Juraschek and H. Will, "Friendly Clustering - A Distributed Approach to Fair Forwarding in Mobile Ad-Hoc Networks," Short Paper for MANIAC Challenge 2009 at PerCom, March 2009. [Online]. Available: http://www.maniacchallenge.org/fuberlin_friendlyclustering_abstract.pdf

[31] M. Güneş, B. Blywis, F. Juraschek, and O. Watteroth, "Experimentation Made Easy," in *Inproceedings of the First International Conference on Ad Hoc Networks*, Ontario, Canada, September 2009.

[32] M. Güneş, F. Juraschek, B. Blywis, and O. Watteroth, "DES-CRIPT - A Domain Specific Language for Network Experiment Descriptions," in *Inproceedings of the International Conference on Next Generation Wireless Systems (NGWS)*, Melbourne, Australia, October 2009.

[33] M. Güneş, F. Juraschek, B. Blywis, Q. Mushtaq, and J. Schiller, "A Testbed for Next Generation Wireless Networks Research," *Special Issue PIK on Mobile Ad-hoc Networks*, vol. IV, pp. 208–212, October-December 2009. [Online]. Available: http://www.reference-global.com/doi/abs/10.1515/piko.2009.0040

[34] B. Blywis, F. Juraschek, M. Güneş, and C. Graff, "MoNoTrac: A Mobile Node Trace Generator," in *8. GI/ITG KuVS Fachgespräch "Drahtlose Sensornetze"*. Technische Universität Hamburg-Harburg, August 2009.

[35] B. Blywis, M. Güneş, F. Juraschek, P. Schmidt, and P. Kumar, "DES-SERT: A Framework for Structured Routing Protocol Implementation," in *IFIP Wireless Days conference (WD'09)*, Paris, France, 12 2009.

[36] B. Blywis, M. Güneş, F. Juraschek, and P. Schmidt, "Implementation of Routing Protocols for Testbeds," in *2009 International Conference on Wireless Communications & Signal Processing (WCSP'09)*, Nanjing, P.R. China, November 2009.

[37] B. Blywis, F. Juraschek, M. Güneş, and J. Schiller, "Design Concepts of a persistent Wireless Sensor Testbed," in *7. GI/ITG KuVS Fachgespräch "Drahtlose Sensornetze"*, 2008. [Online]. Available: http://cst.mi.fu-berlin.de/publications/pdf/2008-Blywis-FGSN.pdf

[38] M. Güneş, B. Blywis, and F. Juraschek, "Concept and Design of the Hybrid Distributed Embedded Systems Testbed," Freie Universität Berlin, Tech. Rep. TR-B-08-10, August 2008. [Online]. Available: ftp://ftp.inf.fu-berlin.de/pub/reports/tr-b-08-10.pdf

[39] M. Güneş, B. Blywis, F. Juraschek, and P. Schmidt, "Practical Issues of Implementing a Hybrid Multi-NIC Wireless Mesh-Network," Freie Universität Berlin, Tech. Rep. TR-B-08-11, August 2008. [Online]. Available: ftp://ftp.inf.fu-berlin.de/pub/reports/tr-b-08-11.pdf

APPENDIX A

Selbständigkeitserklärung

Ich erkläre hiermit, dass

- ich die vorliegende Dissertationsschrift *Distributed Channel Assignment in Wireless Mesh Networks* selbständig und ohne unerlaubte Hilfe angefertigt habe,
- ich mich weder bereits anderwärts um einen Doktorgrad im Promotionsfach Informatik beworben habe noch einen solchen besitze,
- mir die Promotionsordnung der Mathematisch-Naturwissenschaftlichen Fakultät II der Humboldt-Universität zu Berlin (Amtl. Mitteilungsblatt der Humboldt-Universität, Nr. 34/2006) bekannt ist.

Berlin, den 30. Oktober 2013

Felix Shzu-Juraschek