Insights from Experimental Research on Distributed Channel Assignment in Wireless Testbeds

Felix Juraschek Mesut Güneş Matthias Philipp Bastian Blywis
Institute of Computer Science
Computer Systems and Telematics
Distributed Embedded Systems
Freie Universität Berlin, Germany
{jurasch, guenes, mphilipp, blywis}@inf.fu-berlin.de

***Abstract*—This article presents the DES-Chan framework for experimental research on distributed channel assignment algorithms in wireless mesh testbeds. The implementation process of channel assignment algorithms is a difficult task for the researcher since common operating systems do not support channel assignment algorithms out of the box. DES-Chan provides a set of common services required by distributed channel assignment algorithms and therefore eases the implementation effort. Additionally, we present results of experiments to measure the channel characteristics in terms of *intra-path* and *inter-path* interference according to the channel distance on the DES-Testbed. The DES-Testbed is a multi-radio WMN with more than 100 nodes located on the campus of the Freie Universität Berlin. These measurements are an important input to validate common assumptions of WMNs and in order to derive more realistic, measurement-based interference models in contrast to simplified heuristics.**

***Index Terms*—Distributed channel assignment, Wireless mesh network (WMN), Testbed, Experimentation**

I. INTRODUCTION

Channel assignment for multi-transceiver *wireless mesh networks* (WMNs) attempts to increase the network performance by decreasing the interference of simultaneous transmissions. Multi-transceiver mesh routers allow the communication over several wireless network interfaces at the same time. However, this can result in high interference of the wireless interfaces leading to a low network performance. With channel assignment, the reduction of interference is achieved by exploiting the availability of fully or partially non-overlapping channels. Channel assignment can be applied to all wireless networks based on technologies that provide non-overlapping channels. Currently, wide-spread technologies are IEEE 802.11a/b/g, IEEE 802.11n, and IEEE 802.16 (WiMAX). With the relatively low cost for IEEE 802.11 hardware, the number of deployments based on this technology is increasing and channel assignment algorithms are gaining in importance.

Although channel assignment is still a young research area, many different approaches have already been developed [1]. These approaches can be distinguished into *centralized* and *distributed* algorithms. Centralized algorithms rely on a central entity, usually called *channel assignment server* (CAS), which calculates the network-wide channel assignment and sends the result to the mesh routers. In distributed approaches, each mesh router calculates its channel assignment based on local information. Distributed approaches can react faster to topology changes due to node failures or mobility and usually introduce less protocol overhead since communication with the CAS is not necessary. As a result, distributed approaches are more suitable once the network is operational and running. Another classification considers the frequency of channel switches on a network node. In *fast channel switching* approaches, channel switches may occur frequently, in the extreme for every subsequent packet a different channel is chosen. The limiting factor for dynamic algorithms is the relative channel switching time with commodity IEEE 802.11 hardware, which is in the order of milli seconds. *Slow channel switching* approaches in contrast, switch the interfaces to a particular channel for a longer period, usually in the order of minutes or hours. *Hybrid* approaches combine both methods.

An important input for channel assignment algorithms are the particular channel characteristics of the used network technology. For instance, IEEE 802.11b/g offers in theory three non-overlapping channels, e.g. {1, 6, 11}, and all available channels in IEEE 802.11a use non-overlapping frequency spectrums. This means, that concurrent transmissions on these channels should not interfere with each other. In practice, experiments and

---

This work has been partially sponsored and supported by the European Union OPNEX project.

measurement on different experimental platforms have shown, that the non-interfering characteristics do not hold for many reasons [2]–[4], an important one is the unsufficient distance of less than $1\ m$ between the antennas on a single network node. Since mesh routers are usually quiet compact, it is almost impossible to design a multi-radio mesh router with sufficient antenna distance. This is also the case with DES-Nodes of the DES-Testbed, on which the three WiFi antennas are mounted with a distance of about $30\ cm$. Therefore, we also expect to experience side-effects on the theoretical non-interfering channels, which are subject to experimentation in this article.

Next to the channel characteristics of the DES-Testbed, the focus of this article is on the experimentally driven research of distributed, slow channel switching algorithms on wireless testbeds. This process yields several challenges and pitfalls for the researcher. Since common operating systems are not designed to support channel assignment algorithms out of the box. Thus, the researcher has to deal with operating system specifics, drivers for the wireless interfaces, and the capabilities and limitations of the particular hardware. If more than one particular algorithm should be studied, the same problems and services have to be addressed multiple times. Among them are interface management, message exchange for node-to-node communication over the wireless medium, and the provision of data structures for network and conflict graphs.

A research framework for channel assignment algorithms can be beneficial for the implementation process in many ways. The framework introduces an abstraction for low-level and operating system specific tasks, for instance for the configuration of the wireless interfaces. In contrast to the implementation of one specific channel assignment algorithm, a framework should be as universal as possible in order to allow the implementation of a wide range of different algorithms. This is achieved by providing a set of basic services, which are common to multiple channel assignment algorithms.

The contribution of this article is two-fold. On the one hand, we present DES-Chan, a framework for distributed channel assignment in multi-transceiver WMNs that provides a wide range of common services distributed channel assignment algorithms. On the other hand, we present the results of experiments in order to measure the channel characteristics of the DES-Testbed. The gained results will serve as an important input for channel assignment algorithms. Additionally, the results allow to better estimtate the increase in performance of channel assignment algorithms since the interference estimation of two channels is more close to reality than simplified models.

The remainder of this article is structured as follows. In Section II we briefly describe the DES-Testbed which is the foundation for the described experimental research. Section III presents the DES-Chan framework for distributed channel assignment. Section IV presents the experiments and results for the channel characteristics of the DES-Testbed. In Section V we briefly present algorithms for distributed channel assignment and related work of channel assignment frameworks. The article concludes with a discussion of the results and an outlook.

II. DES-TESTBED

The DES-Testbed is a multi-radio WMN located on the campus of the Freie Universität Berlin. Currently it consists of more than 100 indoor and outdoor nodes Figure 2. The hybrid DES-Nodes consist of a *mesh router* and a *sensor node* 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 campus, a snapshot of the network topology is depicted in Figure 1 with the DES-Vis 3D-visualization tool. Besides the DES-Testbed, several in-parallel IEEE 802.11 networks exist to provide network access to students and staff members on our campus. These networks are not under our control and thus contribute to the external interference We treat this as a condition that is also likely to be expected in a real world scenario. For a description of the architecture of the DES-Testbed in full detail we refer to our technical reports [5], [6].

Each DES-Node in the DES-Testbed is equipped with three IEEE 802.11 WNIC. One of the interfaces is a Ralink RT2501 USB stick and the other two are Mini PCI cards with an Atheros AR5413 chipset. The cards use the *rt73usb* and *ath5k* drivers, which are part of the Linux kernel. For the experiments presented in this chapter the Linux kernel 2.6.34 was used. While the Ralink WNIC are IEEE 802.11b/g devices using the $2.4GHz$ band, the Atheros WNIC additionally support the IEEE 802.11a standard on $5GHz$.

Although the 5GHz band theoretically offers 19 non-overlapping channels, only four of these can be used per default in the DES-Testbed. The reason is, that the Atheros cards only support IEEE 802.11a and not the IEEE 802.11h extension which adapts the standard to the European regulatory requirements. Since we are interested in the channel characteristics regardless of a specific regulatory domain, we configured a static regulatory domain database for the Linux kernel and

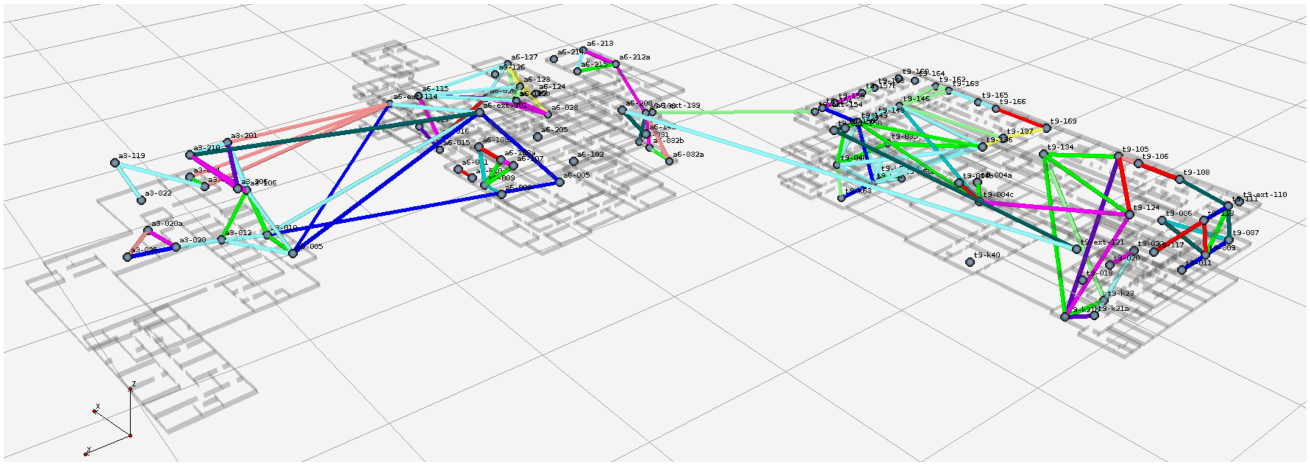

Figure 1. Snapshot of the DES-Testbed topology after a random channel assignment. The DES-Nodes are distributed over three buildings on the campus of the Freie Universität Berlin. Outdoor DES-Nodes are deployed to improve the connectivity between the adjacent buildings. Different colors are used for the different channels of the displayed links. The displayed channel assignment has been calculated with a simple random channel assignment algorithm based on DES-Chan.

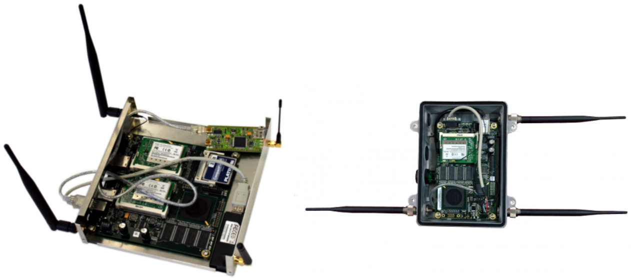

Figure 2. Indoor and outdoor DES-Nodes of the DES-Testbed. The left picture shows the DES-Node version 2. The multi-radio mesh router consists of an Alix2d2 board with three IEEE 802.11a/b/g Ralink- and Atheros-based radios. An additional sensor node is connected to the DES-Node via USB. The outdoor node comprises the same components as the indoor node, but uses the Alix3d2 board to fit into the certified enclosure.

removed all restrictions. Unfortunately, the ath5k driver has a hard-coded limitation for the ad-hoc mode in the upper 5 GHz band which had to be removed as well. As a result, all available 19 channels of IEEE 802.11a can be used for the following experiments on the DES-Testbed.

III. DES-CHAN

The DES-Chan framework has been developed for experimentally-driven research on distributed channel assignment in real network environments. DES-Chan has been implemented as a Python framework and it is available at the website of the DES-Testbed http://www.des-testbed.net. This section presents the requirements for DES-Chan, its components, and the integration into the existing DES-Testbed management system.

A. *Requirements and Services*

A framework for the implementation of distributed channel assignment algorithms should at least meet the following two key requirements. Firstly, it should provide an abstraction layer to the low-level and operating system specific tasks, enabling the researcher to spend most development time on the algorithm logic. Secondly, common key services which are used by multiple algorithms have to be provided.

Naturally, all algorithms need a service for *interface handling* in order to change the wireless network interface card (WNIC) settings, for example to carry out channel switches. Additionally, the local topology has to be assessed with a *neighborhood discovery* service. Information of the local topology is needed as input for channel assignment decisions in distributed algorithms. A *message exchange* service for node to node communication between the instances of an algorithm is useful to exchange topology information messages and to implement a three-way handshake mechanism. A *topology monitoring* service periodically assesses the local topology and its state. Neighbors, links, and their respective quality can be monitored which enables the algorithms to adapt to topology changes by refining the channel assignment.

Interference models are used to estimate the local interference. They range from simple heuristics such as the m-hop neighborhood, where $2 \leq m \leq 3$, to measurement-based approaches for particular network topologies [7], [8]. Even though, the simple heuristics are not very realistic [7], they are easy to implement and therefore widely used. Another common model is the *conflict graph* with algorithms trying to minimize its edges in order to minimize the network-wide interference [9]. Therefore, a framework should provide appropriate data structures and operations for modeling conflict graphs.

B. Architecture and Components

The DES-Chan framework comprises two main components as depicted in Figure 3. *DES-Chan-Core* is a Python library that provides common functions and data structures for channel assignment algorithms. It comprises the following services: *interface management*, *node communication*, *graph representation*, and *interference models*. The *Neighborhood-Discovery* module provides a basic service for each node to get information about all neighboring nodes.

As future algorithms will require additional services, DES-Chan has been designed to be extensible by additional modules, Therefore, a modular architecture has been chosen. As next, the components are described in detail.

1) Interface Management: The interface management module acts as interface to the operating system and hides testbed-specific characteristics from the channel assignment algorithms. It provides various functions for configuring network interfaces and getting information about their state. Thus it is possible to set up and shut down interfaces, check whether an interface has been set up, and get information of unused interfaces. The module furthermore allows to tune an interface to a specific channel and thereby implements a crucial requirement for all channel assignment algorithms. The settings of the wireless interfaces are changed with *Python WiFi* [10], a library that provides read and write access to a wireless network card's capabilities using the Linux Wireless Extensions.

2) Neighborhood-Discovery: The Neighborhood-Discovery module is a daemon that determines the links and their quality on each network node based on the ETX link metric. The ETX link metric estimates how many transmissions for a packet are required so that it is successfully received [11]. ETX values are calculated by each node sending broadcast probes and logging how many probes from their 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

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

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. By default, a probe interval of one second and a window size of 10 seconds is used. The values can be configured via command line arguments. The link quality values are averaged over a moving window, which spans 10 probe intervals. The daemon sends UDP probe packets on the broadcast addresses of the specified network interfaces and at the same time listens for incoming probes. 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.

To offer the neighborhood and link quality information to other programs, `etxd` provides an *inter process communication* (IPC) interface that can be accessed via sockets. A simple, textual protocol allows other applications such as channel assignment algorithms to get the neighbors of a node, as well as the quality and the channel of a certain link. The daemon can be queried to return only those neighbors, which are reached via reliable links, i.e., links whose quality exceeds a certain value. In addition to the link quality, `etxd` also returns the current channel of the WNIC that is used to reach the respective neighbor.

As a result, the ETX daemon provides the neighborhood discovery and the topology monitoring service. Via the IPC interface, channel assignment algorithms can query the current state of their links and react to topology changes.

3) Node Communication: Different channel assignment algorithms use different communication protocols and message formats. Thus, a flexible networking library is needed, that allows implementing various protocols with few effort. The Python Twisted [12] library serves as the foundation for the node to node communication in DES-Chan. It provides an asynchronous networking engine, targeted to ease the implementation of network programs.

The library provides an asynchronous networking engine and hides technical details like creating sockets and establishing connections from the developer. The core of Twisted is a global reactor object that can be instructed to monitor sockets and to connect to servers. The reactor implements an event loop that waits for events, such as an incoming client connection or a certain response from a server, and executes the associated callback functions. In contrast to threaded programming there is no concurrency, because only one function is active at a time. The reactor has control of the single program thread and if an event occurs, it hands the control to

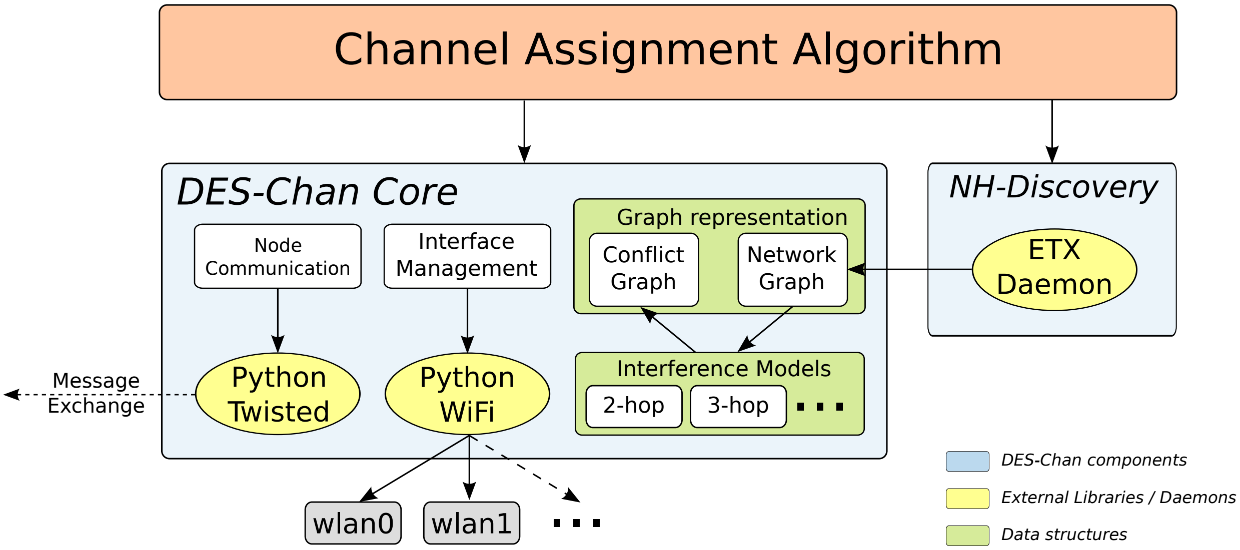

Figure 3. The DES-Chan framework for channel assignment comprises the DES-Chan core and the Neighborhood Discovery service. The DES-Chan Core comprises the wrapper functions to configure the wireless network interfaces and the message exchange with neighboring nodes. The component also comprises common data structures for channel assignment and multiple interference metrics and models. The Neighborhood Discovery service allows to retrieve the local network information periodically and measures the quality of all discovered links using the ETX metric.

the associated callback. When the callback is finished, control is handed back to the Twisted event loop, which continues waiting for the next event.

Using this library, 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.

4) Graph Representation: DES-Chan provides several data structures and functions for graph representation. 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 edges can be labeled with the number of the channel that is used for the communication. The NetworkGraph class also allows to store a list of channels for each edge, thereby supporting multiple links between node pairs. It is limited to undirected graphs, because it has been designed for IEEE 802.11 networks, which only have bi-directional links.

The NetworkGraph class offers intuitive methods for accessing vertices and edges, and provides several utility functions. It supports printing a human-readable adjacency matrix, which can be used for debugging purposes and for saving the graph in a file. Graph objects can also be stored in the *DOT* format, which is a simple textual graph description language that can processed to generate an image of the graph [13]. A function provides the possibility to merge two NetworkGraph objects together by creating the union of both vertex sets and edge sets. This can be used for example to merge network graphs that were obtained from different neighbors.

Additionally, a corresponding ConflictGraph data structure is provided. It allows to apply an interference model to a network graph, in order to calculate the interference between all link pairs in the corresponding network. The class maintains the relation between edges in the network graph and vertices in the conflict graph, and provides corresponding transformation methods. It allows to manipulate the channel of the corresponding edge in the network graph and automatically updates the interference information in the conflict graph. This functionality is needed by algorithms that successively apply several channel assignments to find a configuration that minimizes interference. The ConflictGraph can be easily extended to implement other concepts, such as multi-radio conflict graph [14].

5) Interference Models: The DES-Chan framework supports the implementation of various interference models. Currently, only the two-hop interference model is implemented for reference, but the framework can be easily extended. The current implementation of the two hop heuristic supports a binary notion of interference, i.e., two links either interfere or do not interfere, and a more accurate notion of the interference cost taking the spectral distance of two channels into account.

Equation 2 shows the interference cost function I for two frequencies f_1 and f_2. The additional parameter α denotes the minimum frequency difference of orthogonal channels.

$$I(f_1, f_2, \alpha) = \begin{cases} 0 & \text{if } |f_1 - f_2| \geq \alpha \\ 1 - \frac{(|f_1 - f_2|)}{\alpha} & \text{otherwise} \end{cases} \quad (2)$$

For IEEE 802.11b/g three channels of the 2.4GHz frequency band are theoretically orthogonal. This can be modeled by setting α to 30MHz, which results in the set of orthogonal channels $\{1, 7, 13\}$.

Experiments are presented in Section IV in order to validate the assumption of the orthogonality.

C. Integration into the DES-Testbed

The DES-Testbed Management system has been extended to support the DES-Chan framework [15]. The 3D-visualization tool DES-Vis was modified so that links are colored differently depending on the utilized channel. This way, a graphical representations of the channel assignment can be displayed. A screenshot of an experiment with a random channel assignment shows the capabilities of the visualization tool in Figure 1.

IV. CHANNEL CHARACTERISTICS OF THE DES-TESTBED

This section documents several experiments that have been carried out on the DES-Testbed in order to validate common assumptions of WMN with results on a real multi-radio mesh network. The experimentally determined channel characteristics are an important input for channel assignment algorithms. For instance, co-channel interference can be measured and thus the existence of possible non-interfering channels can be validated. Additionally, the results of the experiments can be used to specify upper bounds for the expected performance increase by channel assignment algorithms.

A. Co-Channel Interference Measurements

One of the proposed measurement-based interference estimation schemes is the *link interference ratio* (LIR) as introduced in [7]. For two links $l_{u,v}$ and $l_{x,y}$ the LIR is defined as

$$LIR_{l_{u,v},l_{x,y}} = \frac{T_{l_{u,v}}^{l_{u,v},l_{x,y}} + T_{l_{x,y}}^{l_{u,v},l_{x,y}}}{T_{l_{u,v}} + T_{l_{x,y}}}$$

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 interference of two links 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.

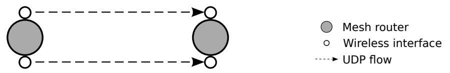

Figure 4. Experiment setup for measuring the effect of spectral channel distance to the LIR with adjacent traffic flows. 5 node pairs of the DES-Testbed are selected to measure the LIR of their corresponding links.

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 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 *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. We repeated the experiment so that we have at least 40 measurements for each channel distance.

1) Adjacent Traffic Flows: First, we need to select a subset of the links between the mesh routers of the DES-Testbed. For this, we used the ETX daemon of the DES-Chan framework to identify high quality links in our testbed with an ETX value of 1. We then selected 5 node pairs which are connected by such high quality links. For each node pair, we selected one node as sender and the other as receiver, as depicted in Figure 4.

In order to measure the LIR we generate two UDP unicast flows from one of the routers (sender) to the other (receiver). Each flow is generated with `iperf` using 54 $MBit/s$ for 30 seconds. After the 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 node on a path forwards traffic to a destination. For simpilicity, we reduced the set up to only two nodes, in which the sender and receiver utilize two radios each. Also, the advantage of multi-radio nodes lies in the capabilities to utlize more than one radio at the time and thus increase the throughput.

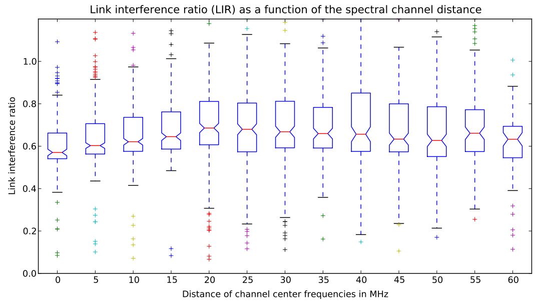

Figure 5. Results of the LIR of adjacent flows for channel combinations on the $2.4\ GHz$ band. The LIR of two links in respect to their spectral distance is shown. The median for all channel combinations is about 0.6.

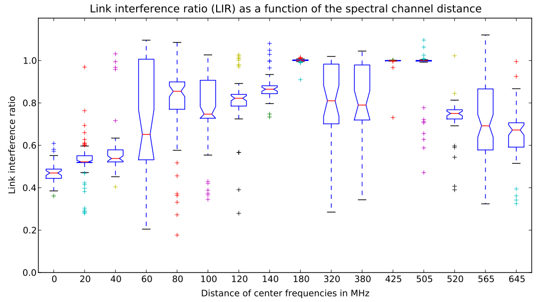

Figure 6. Results of the LIR of adjacent flows for channel combinations on the $5\ GHz$ band. The median of the LIR value increases with channel distances of up to $180\ MHz$ and then decreases again.

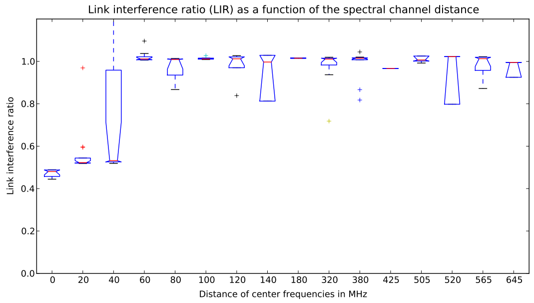

Figure 7. Results of the LIR of adjacent flows on links with a high RSSI value for channel combinations on the $5\ GHz$ band. The average RSSI value for received frames was about $-65\ dBm$. The LIR is close to 1 with a spectral distance of at least $60\ MHz$, which is as expected.

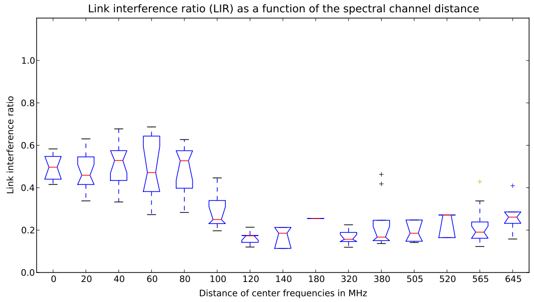

Figure 8. Results of the LIR of adjacent flows on links with a low RSSI value for channel combinations on the $5\ GHz$ band. The average RSSI value for received frames was about $-89\ dBm$. The LIR is less than .5 for channel combinations with a spectral distance of more than $100\ MHz$. We assume the low RSSI values to cause this effect, which we will further investigate in future work.

The results for the channel combinations on the $2.4\ GHz$ band are depicted in Figure 5. Unfortunately, they show that in the DES-Testbed none of the channels of the $2.4\ GHz$ band are non-interfering. 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 the two links are active simultaneously. Concluding from the results, a channel assignment with the highest possible spectral distance would only lead to a minor increase of the throughput. We credit these results to the near-antenna effect of the DES-Nodes. To avoid the near-antenna effect, the experimentally specified minimum distance between two antennas is about $1\ m$ [16]. Since DES-Nodes are more compact, the three WiFi antennas are mounted with a distance of about $30\ cm$.

The results for a subset of all possible spectral channel distances of the $5\ GHz$ band are depicted in Figure 6. 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 expect. In a first investigation, this seems to be related to the link quality. For the UDP flows, we monitored the *received signal strength indicator* (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.

To display the results, we included the measured LIR values of two different node pairs u, v and j, k in Figure 7 and Figure 8. For the first node pair, the measured LIR values increase with a rising channel distance, which is as expected. We measured an average of about $-65\ dbm$ for all received frames. For the second node pair, the measured LIR values behave unexpectedly and start to drop already at a channel distance of $100\ MHz$ below 0.5. We measured an average of about $-89\ dbm$ for all received frames, which is close to the threshold of the WNIC being able to receive a frame correctly. As a conclusion of this observation, we suspect that the huge difference in the RSSI values does hint at a very different link quality of these two link

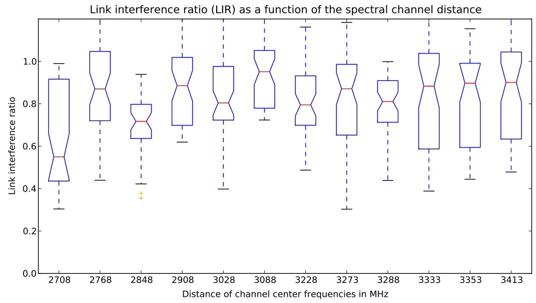

Figure 9. Results of the LIR of adjacent flows for channel combinations on the 2.4 GHz and 5 GHz band. The median of the LIR for all channel combinations is about 0.8. This means that the links only exert minor interference effects on each other.

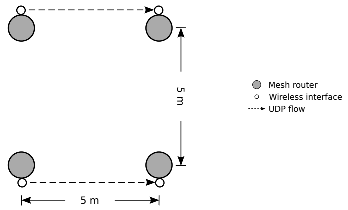

Figure 10. Experiment setup for measuring the effect of spectral channel distance to the LIR with non-adjacent traffic flows. Two node pairs are selected which are located in the same room. The LIR of the two links between the node pairs is measured.

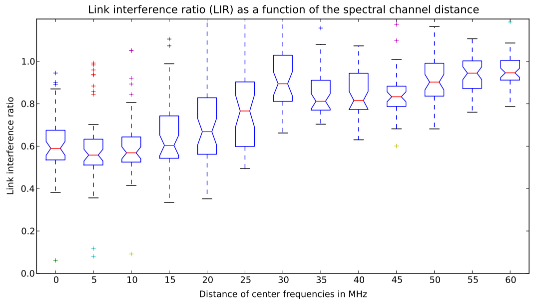

Figure 11. Results of the LIR of non-adjacent flows for channel combinations on the 2.4 GHz band. With a channel distance of 30 MHz, the median of the LIR is usually above 0.8 which implicates that a significant higher throughput can be achieved.

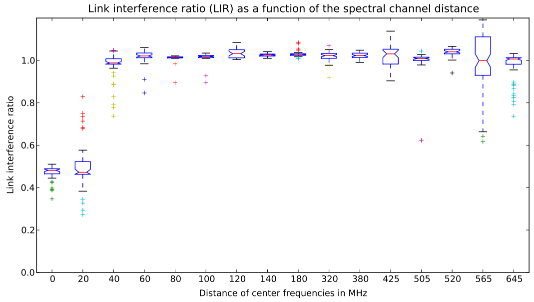

Figure 12. Results of the LIR of non-adjacent flows for channel combinations on the 5 GHz band. From a channel distance of 60 MHz, the median of the LIR is close to 1, which means that there are hardly any interference effects.

pairs. Unfortunately, the broadcast-based ETX-daemon does not seem to be appropriate to estimate link quality for unicast transmissions. We will further investigate this observation in future work.

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 WNIC 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 9, show that the median of the LIR is between 0.8 and 1 and therefore only a small decrease of performance can be observed.

Unfortunately, the results of the experiments with adjacent flows 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 only 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.

2) Non-Adjacent Traffic Flows: In the second experiment 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 10.

The results for the channel combinations on the 2.4 GHz band are depicted in Figure 11. 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 at least that distance.

The results for the 5 GHz band are depicted in Figure 12. 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.

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 13, show that the median of the LIR is close to 1 for all channel combinations and therefore no interference effects are observed.

Although none of the channel combinations in the 2.4 GHz 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

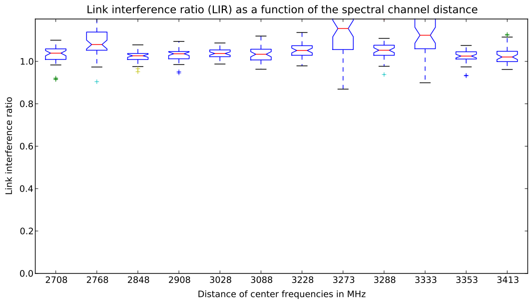

Figure 13. Results of the LIR of non-adjacent flows for channel combinations using the 2.4 GHz and 5 GHz band. The median of the LIR is close to 1 for all channel combinations and therefore no interference effect is observed

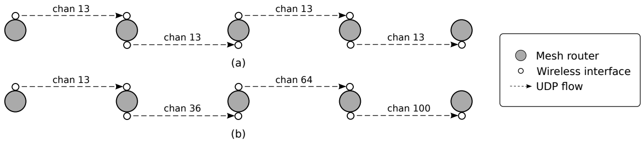

Figure 14. Experiment setup to measure throughput in a single- and multi-channel network. A subset of the mesh routers is selected to create a chain topology. In (a), we apply the same channel to all wireless links, thus creating a single channel network scenario. Based on the results of the previous experiments, we manually apply channels to the links which promise an increase of throughput compared to the single network scenario in (b).

assignment. On the 5 GHz band a minimum spectral channel distance of 40 MHz is sufficient to experience only neglectable interference effects. Using both bands simultaneously, hardly any interference effects could be measured with the median of the LIR being close to 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 bigger antenna distance for the experiments with non-adjacent flows (5 m to 0.3 m). To validate the assumption, we will perform experiments with adjacent flows and longer antenna cables therefore increasing the inter-antenna distance in future work.

B. 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 five mesh routers, on which we can 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 scenario as depicted in Figure 14 (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, until the last node in the chain. We configured the WNIC 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.

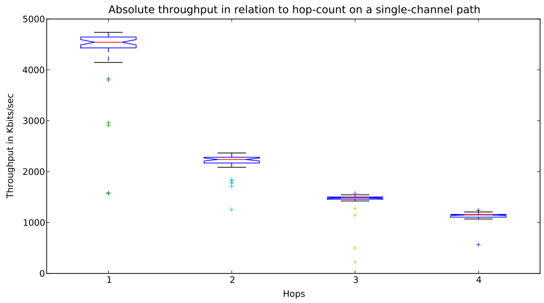

Figure 15. Results for the single-channel path experiment. As expected for the single channel network case, the throughput is more then halved on the first hop and keeps dropping with an increasing hop-count.

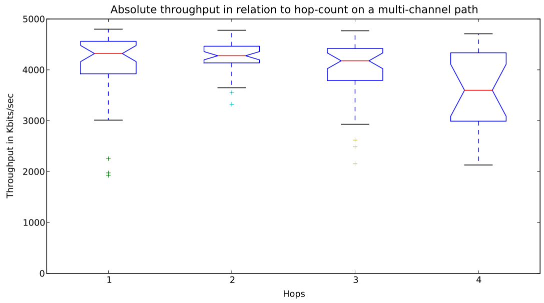

Figure 16. Results for the multi-channel path experiment. As expected, the manual channel assignment lead to a higher throughput if the hop-count is bigger than 1. It only drops slightly with the increasing hop-count which implicates that the interference effects have been reduced significantly with the chosen channel assignment.

The results for the single channel network scenario for the throughput in relation to the hop count are depicted in Figure 15. 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 to the chain topology 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 WNIC. 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 14 (b) with which we expect to exhibit only minor interference.

The results for the multi-channel network scenario are depicted in Figure 16. 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 proper channel assignment.

V. RELATED WORK

Many approaches for distributed channel assignment have already been proposed [1]. In the greedy channel assignment algorithm by Ko et. al [17], one wireless interface of each node is switched to a common channel in order to ensure the network connectivity. For additional interfaces, a greedy algorithm selects the least interfering channel in the interference set using an interference cost function which takes the degree of overlap of two channels into account. The *distributed greedy algorithm* (DGA) assigns channels to links instead of interfaces and is therefore topology preserving, meaning that all links are sustained during the channel assignment procedure [3]. The link-based channel assignment approach by Sridhar et. al is similar to DGA with the difference that it additionally takes the expected traffic-load on the nodes into account [18]. Interference is modeled with a fixed interference range for all nodes and a weighted conflict graph is used to estimate the network-wide interference. The *Skeleton Assisted Partition Free* (SAFE) algorithm uses *minimal spanning trees* (MSTs) to preserve the network connectivity [19].

A fast channel switching algorithm is proposed by Kyasanur et. al in [20], [21]. The set of network interfaces on each node are divided into *fixed* interfaces, which stay on a fixed channel, and *switchable* interfaces. 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 crucial part of this approach is the way how channels are assigned to the fixed interfaces. The NET-X framework was created to implement the algorithm in a wireless testbed environment based on the 2.4 Linux kernel [22]. The implementation of this particular algorithm requires several changes to the Linux network stack and to the driver of the wireless network interfaces. The services provided by NET-X are directly derived from the requirements of the implemented algorithm [21]. Slow channel switching algorithms can also be realized with NET-X, although it is not in the focus and to the best of our knowledge, none has been implemented so far. Recently, experiments with the NET-X framework are performed to evaluate algorithms for assigning channels to the fixed interfaces [23] and for QoS-provisioning, extensions have been developed for queue management [24]. The framework is not intended to as foundation to delevop different 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.

VI. DISCUSSION AND OUTLOOK

With DES-Chan a wide range of different algorithms can be implemented, validated, and compared in a real network environment. DES-Chan does not require any changes of the Linux kernel or the wireless network interface drivers. It is therefore easy to integrate into existing wireless mesh network testbeds. The contribution of DES-Chan is two-fold. Firstly, DES-Chan provides an abstraction layer to the low-level and operating system specific tasks. This abstraction layer enables the researcher to spend most development time on the algorithm logic. Secondly, DES-Chan provides basic services and data structures often required for typical tasks in channel assignment algorithms. For instance, appropriate data structures have been provided for network graphs, conflict graphs, and interference models. Currently, we are working on an implementation of a link-based channel assignment similar to DGA [3].

The basic experiments and measurements were performed in order to gain insights on the network topology and the channel characteristics in the DES-Testbed. In a first series of experiments, the effects of the co-channel interference have been investigated. The LIR of two links is significantly lower for adjacent than for non-adjacent traffic flows. Nevertheless, 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 is significantly higher using the manual channel assignment 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 are transferred to the channel assignment algorithms, the actual number of available channels is significantly reduced which may affect the performance of the algorithms. Second, the experiment results for adjacent and non-adjacent flows show different characteristics. Therefore, channel assignment algorithms should distinguish between adjacent and non-adjacent flows to optimally assign the available channels.

Future work will be focused on research of more realistic, measurement-based interference models. With these models, it is likely to close the gap between the graph-based and throughput-based results obtained from the experiments on the DES-Testbed. Additionally, we will implement further algorithms based on DES-Chan.

REFERENCES

[1] 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. In Press, Corrected Proof, pp. –, 2009. [Online]. Available: http://www.sciencedirect.com/science/article/B6WKJ-4XCJ4MF-2/2/695f3370fc36df4a5b96b0de4d2782c1

[2] 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.

[3] 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.

[4] 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.

[5] M. Günes, B. Blywis, and F. Juraschek, "Concept and Design of the Hybrid Distributed Embedded Systems Testbed," Freie Universität Berlin, Tech. Rep. TR-B-08-10, 2008.

[6] M. Günes, 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, 2008.

[7] 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*. Berkeley, CA, USA: USENIX Association, 2005, pp. 28–28.

[8] 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.

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

[10] R. Joost and S. Robinson, "pythonwifi," http://pythonwifi.wikispot.org/, Last visit: 2011.

[11] 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.

[12] *The Python Twisted Documentation*, Last visit: 2011. [Online]. Available: http://twistedmatrix.com/projects/core/documentation/howto/book.pdf

[13] G. Project, "Graphviz project," 2010. [Online]. Available: http://www.graphviz.org/

[14] 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

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

[16] 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.

[17] B.-J. Ko, V. Misra, J. Padhye, and D. Rubenstein, "Distributed channel assignment in multi-radio 802.11 mesh networks," 2007, pp. 3978–3983. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4204903

[18] S. sridhar, J. Guo, and S. Jha, "Channel assignment in multi-radio wireless mesh networks : A graph-theoretic approach," 2009.

[19] 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*, vol. 0, pp. 417–426, 2006.

[20] 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

[21] 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.

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

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

[24] 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.