Novel Metrics and Experimentation Insights for Dynamic Frequency Selection in Wireless LANs

Giannis Kazdaridis Stratos Keranidis Adamantios Fiamegkos
Thanasis Korakis Iordanis Koutsopoulos Leandros Tassiulas

Department of Computer and Communication Engineering, University of Thessaly, Greece
Centre for Research and Technology Hellas, CERTH, Greece
E-mail: {iokazdarid, efkerani, adfiameg, korakis, jordan, leandros}@uth.gr

ABSTRACT

The rapidly increasing popularity of IEEE 802.11 WLANs has created unprecedented levels of congestion in the unlicensed frequency bands, especially in densely populated urban areas. Performance experienced by end-users in such deployments is significantly degraded due to contention and interference among adjacent cells. In this paper, we develop novel metrics and insights that we use for dynamic frequency selection, incorporating the various features that affect interference. The proposed scheme features a novel client feedback mechanism, which enables nodes of the cell, as well as nodes belonging to different cells, to contribute to interference measurements. Furthermore, we incorporate a traffic monitoring scheme that makes the system aware of prevailing traffic conditions. We design a distributed protocol, through which messages containing the information above are passed by the stations to the access-points, where the frequency selection is performed in a dynamic form. The proposed algorithm is implemented in the Mad-WiFi open source driver and is validated through extensive testbed experiments in both an indoor RF-Isolated environment, as well as in a interference-rich, large-scale wireless testbed. Results obtained under a wide range of settings, indicate that our algorithm improves total network throughput, up to a factor of 7.5, compared to state-of-the-art static approaches.

Keywords

Wireless communications, Dynamic Frequency Selection, Experimentation, MAC, IEEE 802.11

1. INTRODUCTION

The tremendous growth of 802.11 WLANs has resulted in congestion of the limited unlicensed spectrum, especially in densely populated urban areas. Moreover, the uncoordinated management of WLANs, in accordance with the limited number of non-overlapping channels, leads to increasing contention and interference conditions. Consequently, the throughput performance experienced by wireless stations (STAs) is significantly degraded.

In infrastructure 802.11 WLANs, channel selection is performed at the AP. In most cases, the operational frequency is configured through manual input upon network initialization. Another common approach followed state of the art approaches is that of selecting the channel that offers the lowest received signal strength [1]. Moreover, some vendors, based on traffic measurements, aim to avoid frequencies that are highly congested by nodes belonging to other networks [2]. Such approaches result in static channel assignments, a scheme that is not consistent with the dynamic nature of the wireless medium.

In this work, we propose a frequency selection algorithm that dynamically switches the operational channel by taking into account several factors that affect end-user performance.

1.1 Related Work

Initial efforts on channel assignments for wireless networks date back to the 1980s [3]. In addition there exist several approaches on frequency selection in infrastructure WLANs that have been proposed in the recent literature. Initial attempts in the field, aiming to minimize interference caused by channel and cell overlapping are usually formulated in terms of graph coloring theory. In [4], the authors construct an interference graph that requires exchange of information among APs, in order to estimate cell interference. The work in [5] proposes a distributed interference mitigation mechanism restricted in the case of non-overlapping channels only. Another work proposed in [6], suggests that client feedback is important in detecting APs that appear hidden to the corresponding APs. However, this approach considers only fixed locations for both the APs and their associated STAs. The work proposed in [7] advocates a novel mechanism based on the notion of channel-hopping that improves fairness in distribution of throughput among cells. The approaches above, share the common characteristic of considering that all network nodes exhibit constant traffic activity at all times.

However, since the assumption of constant traffic conditions is unrealistic, channel selection should also take into account the load of each node that potentially causes interference. Trying to address this issue, the authors in [8] incorporate traffic demands of both APs and STAs in the proposed mechanism. Nevertheless, this approach was restricted in considering uniform demands across all associated

WiNTECH'11, September 19, 2011, Las Vegas, Nevada, USA.

STAs of a specific AP. A channel selection mechanism that considers channel conditions as well as AP load was proposed in [9]. Although this approach is load-aware as well and thus offers significant improvement, it considers only transmissions of associated STAs in the definition of load, ignoring other neighboring nodes that contend for channel occupancy.

Channel assignment has also been considered in the context of multi-channel multi-radio Wireless Mesh Networks. In [10], the proposed mechanism assigns channels to multi-radio nodes with the overall goal of ensuring that neighboring nodes are able to communicate successfully. The work in [11] studies the problem of frequency selection and routing jointly, by capturing the end-to-end link loads across different routes.

Most of the works referenced above rely only on simulation based evaluation of the proposed algorithms. In [4], a rather simple implementation is presented that requires a wired distribution system to operate. The work in [8] includes only limited testbed results, without providing any issues related to the implementation efforts. On the other hand, the work in [6] implements a simplified version of their algorithm, using precomputed range and interference sets in the experimental evaluation. The work in [7] provides an application-level implementation of the proposed channel hopping mechanism. Nevertheless, we argue that frequency selection algorithms for 802.11 WLANs, should be fully implemented and properly evaluated through extensive experimentation in real world network scale and settings in order to conclude on realistic results in real interference and congestion conditions.

1.2 Our Contribution

In this paper, we study dynamic frequency selection in infrastructure 802.11 WLANs. The key novelty of our approach is that we incorporate different, hitherto unexploited features affecting total network interference and we devise a distributed protocol suite that dynamically selects the operating channel. We start by considering the degree of overlap among adjacent channels based on a set of extensive experiments, designed to estimate the impact of interference on throughput performance, under various settings.

As a second contribution, we incorporate a client-assisted mechanism to enable the STAs of each Basic Service Set (BSS) to participate in interference estimation. Moreover, unlike relevant approaches, we extend our client feedback mechanism, so that each AP can further utilize interference measurements reported by STAs belonging to other BSSs. Based on this cooperative mechanism, the APs obtain more accurate results and moreover they manage to detect adjacent interfering BSSs that cannot be sensed directly by the AP.

Another key contribution of the proposed scheme is its ability to adapt to varying traffic conditions. Contrary to other approaches, we do not only take into account the traffic rate at which packets are transmitted, but we properly combine the rate at which packets are generated by all transmitters in range, with the corresponding transmission rate used at the physical (PHY) layer, as well as the actual frame size, in order to calculate the occupancy ratio of each channel. Thus, we are able to estimate the level of contention on each available channel.

Our mechanism results in a dynamic algorithm that selects the operating channel dynamically, properly considering the fundamental issues that affect the end-user performance. Finally, another important contribution is that we move one step further from simulation approaches and implement the proposed algorithm using the Mad-WiFi open source driver [1]. We conduct extensive experiments in both RF-isolated as well as interference-rich environment, in order to evaluate the performance in real world settings. To the best of our knowledge, this is the first work in the literature, featuring a complete driver-level implementation, accompanied with such extensive experimental results.

Table 1: Measured I_{factor} - Channel Separation

Channel Separation ($\|m-n\|$)	0	1	2	3	4	5	6
Measured I_{factor}	1	0.75	0.37	0.1	0.02	0	0

The rest of the paper is organized as follows. A detailed analysis of the interference model followed is presented in section II. In section III, we describe an initial set of experiments that motivated our work. Details about the metrics used in our approach and the corresponding implementation are provided in section IV and V. The configuration of our experiments, concerning the testbed and the methodology used, is then described in section VI. In section VII, we experimentally evaluate the performance of our implementation. Finally, in section VIII, we present the conclusions and discussion of future work.

2. INTERFERENCE MODEL

2.1 Overlapping Channels Interference

IEEE 802.11 set of standards make use of the ISM (Industrial Scientific Medical) bands. The popular 2.4 GHz band, used by 802.11b and 802.11g standards, offers 11 consecutive channels, spaced 5 MHz apart and occupying 22 MHz of bandwidth. As a result, most channels partially overlap with adjacent cells, limiting the number of non-overlapping channels to three (e.g. 1, 6, 11). Consequently, transmissions on a specific channel may interfere with simultaneous transmissions on overlapping channels.

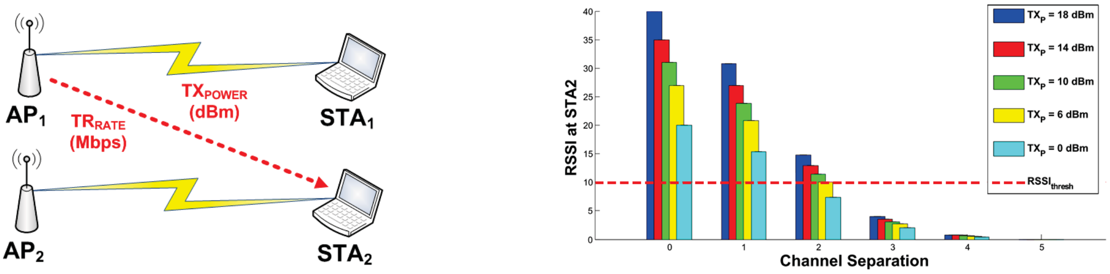
In our work, we use the notion of I_{factor}, introduced in [12] to model the degree of overlapping between transmissions on two certain frequencies. More specifically, we use $I(m, n)$ to quantify the degree of overlap in signal power among the reception of a frame on channel m and the reception of the same frame on channel n. In essence, if $RSS_j(n)$ is the received signal strength (RSS) for a specific frame transmitted by node j on channel n, then the RSS for the same frame on channel m can be estimated as:

$$RSS_j(m) = RSS_j(n) * I(m, n) \qquad (1)$$

Table I lists the measured values of I_{factor} for various channel separation values ($|m - n|$), as used in [12] and it is assumed to be zero for channel separation greater than 6.

2.2 Contention

In the previous subsection, we describe why interference exists among adjacent channels in 802.11. However, interference may also exist in the case that the two nodes operate on the same channel. We denote by TX_{range}, the range (with respect to the transmitter) within which a transmitted frame can be successfully decoded. Moreover, we use CS_{range} to denote the range (with respect to the transmitter), which includes nodes that are able to sense its ongoing transmis-



(a) Experimental Setup.

(b) RSSI - Channel Separation.

Figure 1: Motivating Experiment.

sions and thus defer their own. The IEEE 802.11 medium access is performed by the distributed coordination function (DCF), which is based on the CSMA/CA protocol. As long as transmitter nodes exist in the CS_{range} of each other, the transmissions are mutually detected and channel contention is performed by the collision avoidance mechanism that 802.11 standard supports. As a result, every node in the network shares the medium with its neighboring nodes. The performance decrease observed in this case is caused due to the sharing of the channel and not due to existence of interference.

2.3 Co-channel Interference

In the case that the distance between the communication pair exceeds CS_{range}, there may exist co-channel interference. The range (with respect to the transmitter node) within which interference phenomena may occur, is denoted by IF_{range}. IF_{range} describes the range within which nodes cannot sense the ongoing transmissions of the transmitter, and as a result they can start transmitting simultaneously. These transmissions are subject to frame errors due to interference at the corresponding receivers, which leads to decreased packet delivery ratio (PDR).

3. MOTIVATING EXPERIMENT

In this section, we seek to investigate the impact of interference on throughput performance. We start with the experimental setup shown in Fig. 1(a), using two pairs of nodes, operating on channel 1 of the 2.4 GHz band. The receiver nodes (STAs) of each communicating pair are placed in the TX_{range} of the relative transmitters (APs). In this experiment, we use the AP1 of the upper pair as the interfering transmitter and measure performance at the STA2 of the lower pair.

More specifically, we design our initial set of experiments, in order to demonstrate how interfering signals of varying RSS affect throughput. In this perspective, we decide to vary the transmission power (TX_{power}) of AP1, as well as the channel separation between the two pairs, by switching the operating channel of the second pair between channels 1 to 6. To calculate RSS among overlapping frequencies, we use the I_{factor} values used in [12], as presented in Table I. Fig. 1(b) illustrates how the average RSS measured at STA2 changes with respect to the TX_{power} of AP1 and the varying channel separation. At this point, it is important to note that transmissions received with RSS lower than a certain threshold do not affect the throughput performance of the affected links. Having determined this threshold equal to 10 points in the RSSI scale, we present it with a dashed red line in Fig. 1(b), while Fig. 2(a) represents the throughput results obtained from this first set of experiments.

Our second set of experiments is designed to demonstrate how the varying traffic activity of interfering links affects throughput performance. In this experiment, we keep the transmission power of AP1 constant and equal to the maximum value of 18 dBm, while the traffic rate (TR_{rate}) of the generated flow is varied for the various channel separation values. As clearly shown in Fig. 2(b), there exists clear relation between the activity of interfering links and the corresponding throughput at the affected nodes. These observations lay the motivation for building an analytical model that captures the combined effect of partial overlap among adjacent channels, as well as the varying traffic activity of nodes in range.

4. PERFORMANCE METRIC

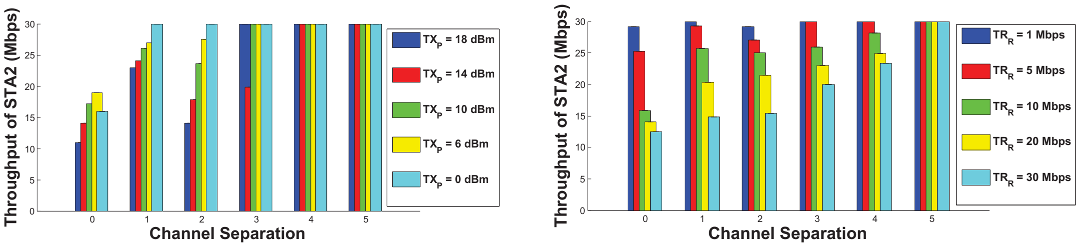
As demonstrated through the motivating experiment, a suitable metric for frequency selection has to consider both interference as well as traffic conditions, in order to be able to define channel assignments that maximize the throughput.

Considering the first requirement, we must ensure that adjacent BSSs are assigned different channels. As long as the nodes of the BSS under consideration are located inside the CS_{range} of nodes belonging to adjacent BSSs, contention and overlapping channels interference phenomena may occur. At first, we suppose that measurements are taken only by the AP. We use $\mathcal{B}$ to denote the set of interfering nodes that can be detected by the AP and $\mathcal{C}$ to denote the set of available channels. The channel m that is selected is the one such that the quantity:

$$\sum_{j \in \mathcal{B}} RSS_j(n) * I(m,n), \tag{2}$$

is the smallest among candidate channels, where we use the notion of I_{factor} to estimate the RSS of frames transmitted by each user $j \in \mathcal{B}$ operating on channel n , as if they were taking place on channel m, which is the channel under consideration.

In order to satisfy the second requirement, we need to estimate the level of congestion a node experiences on each channel. To this end, we introduce the notion of Channel Occupancy Time (COT), which is estimated by the AP that monitors the number, corresponding frame size and PHY rate of detected frames during intervals of fixed duration. The COT for a certain channel m can be calculated, as follows:



(a) Throughput - Transmission Power of AP1. (b) Throughput - Traffic Rate of AP1.

Figure 2: Throughput Performance of Motivating Experiment.

$$COT(m) = \sum_{k=1}^{F_m} \frac{L_k}{R_k}, \tag{3}$$

where we use F_m to denote the number of Monitored Frames on channel m during each such interval, L_k to denote the size of frame k expressed in bit values and R_k to denote the PHY rate used by the transmitter during its transmission.The probability that frames transmitted in adjacent cells interfere with transmissions of our AP, is proportional to their frequency of occurrence as well as their transmission duration. To model this effect, we use COT as a weighting factor, so that each channel is further characterized by the prevailing traffic conditions. We import this effect in expression (2) and select the channel m so that the quantity:

$$\sum_{j \in \mathcal{B}} RSS_j(n) * I(m,n) * COT(m) \tag{4}$$

is the minimum, among candidate channels. As described above, our algorithm supports feedback from both associated STAs, as well as from STAs that belong to other adjacent BSSs. We use $\mathcal{A}$ to denote this set of nodes that provide feedback, by transmitting measurement frames. As long as the AP exists inside the TX_{range} of any node $i \in \mathcal{A}$, it decodes the received frames and benefit from the contained information. Finally, the AP calculates the average metric value, over the total number of nodes providing measurements and selects the channel m, for which quantity:

$$\frac{1}{|\mathcal{A}|} \sum_{i \in \mathcal{A}} \sum_{j \in \mathcal{B}} RSS_{ij}(n) * I(m,n) * COT(m) \tag{5}$$

is minimum, where $RSS_{ij}(n)$ denotes the RSS estimation provided by node $i \in \mathcal{A}$, regarding node $j \in \mathcal{B}$. We have to mention here that the feedback mechanism enables the AP to detect nodes that cause co-channel interference as well, which is not feasible if measurements are taken solely by the AP. We decided to average the results collected at the AP, in order to minimize the estimation error in the metrics calculation, based on the fact that in real environments RSS is considered a random variable, due to shadowing effects.

5. PROTOCOL DESCRIPTION

IEEE has specified the 802.11h [13] amendment that provides for a DFS mechanism in WLANs. However, the problem it addresses is interference with satellites and radars using the 5 GHz frequency band. The algorithm proposed in this work addresses the problem of frequency selection and, like most relevant schemes proposed in the literature, it consists of three main steps, namely *Channel Measurement*, *Channel Selection* and finally the *Channel Switch* procedure.

Regarding the first procedure, in order to gather information about conditions on channels different from the one in use, we have to switch the operating channel to the new one and monitor it for a constant period of time. This procedure is common in infrastructure 802.11 WLANs, where STAs perform *scanning* to discover available APs in range and therefore decide to join a specific BSS. Moreover, STAs perform *scanning* in order to search for potential hand offs. This last procedure is referred to as *Background Scanning* (BG_{scan}), as it is performed seamlessly without resulting in loss of connectivity. The proposed protocol suggests that APs perform BG_{scan} as well to collect *Beacon* frames and estimate levels of interference on each channel. Having collected measurements about the current channel, the AP continues the BG_{scan} by switching between the rest available channels. On the other hand, measurement of traffic conditions as suggested in our work, requires that all transmitted frames are monitored. The APs calculate the COT value related to transmissions belonging to its own BSS, based on equation 3. and then piggyback this information in their *Beacon* and *Probe-Response* frames. At this point, all nodes that perform BG_{scan} manage to estimate performance on each channel, by calculating the metric value described in eq. 4. Moreover, in the perspective of cooperation, we propose a novel client feedback mechanism that enables all the STAs of the BSS to contribute to the discovery phase, by passing special broadcast frames containing their respective measurements to the AP.

After completing the *Channel Measurement* procedure, the AP proceeds with the *Channel Selection* procedure, where it combines its own measurements with measurements gathered from both associated STAs as well as nodes belonging to other BSSs, using eq. 5. Finally, the AP selects the channel offering the lowest calculated metric value. In case the selected channel is different from the one currently used, the AP continues with the *Channel Switch* procedure.

For the third procedure, the proposed protocol follows the standard *Channel Switch Announcement* (*CSA*) procedure, proposed in the 802.11h amendment. According to the standard, the AP informs its STAs for the intending channel switch using the *CSA* frame, which contains the selected channel and the interval after which the switch will occur. The steps required for the completion of the proposed protocol are as follows:

- **STEP 1:** Periodical calculation of COT by APs and piggybacking in *Beacon* and *Probe-Response* frames.
- **STEP 2:** Periodic repetition of BG_{scan} by both STAs and APs, to gather information about interfering BSSs.
- **STEP 3:** Broadcasting of measurements by the STAs.

- **STEP 4:** Collection of measurements by the APs.
- **STEP 5:** Calculation of average metric values per channel at the AP by considering:
    1. AP BG_{scan} measurements
    2. measurements of associated STAs
    3. measurements of neighboring STAs of other BSSs.
- **STEP 6:** Selection of the channel that offers the lowest calculated value.
- **STEP 7:** Broadcasting of *CSA* frame to advertise channel switching, in the case that the selected channel is different from the one currently in use.
- **STEP 8:** Switching to the new channel after a specific interval, defined in the CSA frame.

The steps described run in a dynamic manner, so that channel transitions are continuously performed. In the following section, the key challenges encountered in the driver implementation as well as the corresponding solutions are described.

6. IMPLEMENTATION DETAILS

For the implementation of our mechanism, we used the Mad-WiFi open source driver. A simple static frequency selection mechanism is implemented in the original driver, which suggests that APs, upon initialization of the BSS, perform *scanning* to select the channel that offers the least amount of RSSI. As the already implemented procedure did not fit our needs, we modified the driver to integrate the BG_{scan} feature in the *AP* mode of operation. Moreover, we further modified the BG_{scan} procedure, in the way that not all of the channels are sequentially scanned. More specifically, the complete channel list is scanned in batches of four channels at most. Following this approach, the maximum interval of operation discontinuity is significantly reduced.

In order to carry out traffic estimation, we implemented the COT calculation procedure in the AP mode. However, in this mode, all packets received by the network adapter are filtered out, so that the ones with a destination address different than the local MAC address of the adapter are discarded. Only unicast packets that are destined to the adapterâĂŹs MAC address as well as multicast and broadcast packets can be captured. Instead of putting an additional wireless interface to each node to operate in monitor mode and collect measurements, we implemented an innovative mechanism that exploits the inherent characteristic of infrastructure WLANs. The characteristic that our mechanism exploits is that all the frames transmitted in a certain BSS are either transmitted from the AP, or are destined to it and, as a result, APs are able to estimate traffic conditions in their own BSS accurately. More specifically, each AP manages to calculate the COT value related to transmissions belonging to its own BSS, by recording the length of each frame and the PHY rate used for its transmission or reception. By exchanging such information among adjacent BSSs, traffic estimation per channel can be achieved. To this end, we extended the *Beacon* and *Probe-Response* frames to carry this information. This frame extension does not affect the normal operation of the 802.11 protocol, as these frames feature a dynamic part that supports extension, according to the standard.

The additional feature that enables STAs belonging to different BSSs to contribute to the measurement procedure is not supported by 802.11 compliant WLANs, because such packets are discarded as previously described. To propagate these measurements through different BSSs, we further modified the driver to generate an 802.11 broadcast frame. This special control packet, called *Measurement Report*, is periodically transmitted by each associated STA. The format of the *Measurement Report* frame, as well as the *Beacon* frame extension, are described in Fig. 3.

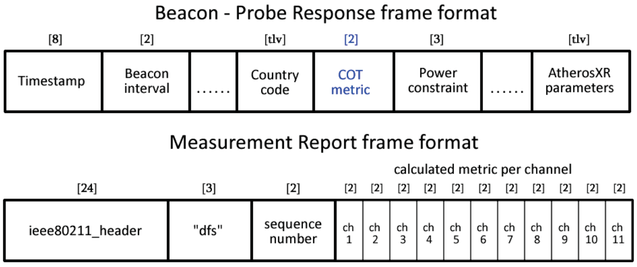


Figure 3: Frames format.

Information gathered from neighboring STAs is properly considered, so that measurements regarding the channel currently in use are discarded as inappropriate, because they consider our own BSS as interference. Moreover, *Beacon* frames transmitted by the AP of our BSS are excluded from the above calculations, as this node should not be considered as an interfering one.

7. EXPERIMENTAL CONFIGURATION

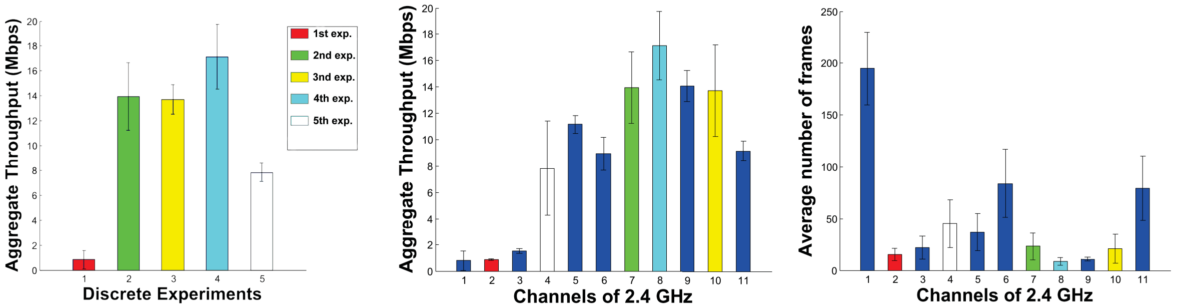
In order to evaluate the performance and study the implemented DFS scheme under realistic conditions, we used a large scale wireless testbed, called NITOS [14].

7.1 NITOS Testbed

NITOS currently consists of 40 wireless nodes, deployed outdoors at the exterior of the University of Thessaly campus building. The nodes are equipped with 2 wireless interfaces using Wistron CM9 - mPCI Atheros 802.11a/b/g 2.4 and 5 GHz cards that run Mad-WiFi open source driver. As NITOS is a non-RF-isolated wireless testbed, we could conduct our experiments under rich interference conditions. More particularly, during the experimentation process we observed the existence of more than 100 interfering APs operating on different channels. This fact gave us prolific settings to prove the validity of our DFS algorithm.

7.2 Measurement Methodology

The throughput performance of the experiments is measured by using Iperf. A typical experimental setup for experiments considering only downlink transmissions, would be to run an Iperf client at the nodes, that act as APs, in order to generate traffic streams, having an Iperf server residing on each STA, receiving the traffic and collecting the measurements. To remove any random effect and short-term fluctuation, we run each experiment 10 times and each run lasts for 10 minutes. In the figures that follow, we use error bars to represent deviation among the multiple executions of the same experiment and average values as the final results.



(a) Throughput - Discrete Experiments. (b) Throughput - Channels of 2.4 GHz. (c) Monitored frames - Channels of 2.4 GHz.

Figure 4: First set of Experiments - Downlink Outdoor Experiments in the 2.4 GHz band.

8. EXPERIMENTAL EVALUATION

The first two sets of experiments have been executed in NITOS using only channels of the 2.4 GHz and 5 GHz bands accordingly, while the third set of experiments was conducted in an indoor testbed. In order to compare performance, we use the unmodified driver, which upon the initialization of the BSS selects the channel that offers the least amount of RSSI based on AP measurements. To demonstrate that our algorithm is rate adaptive as well, we decided to enable the driver Rate Adaptation algorithm, instead of conducting fixed-rate experiments.

8.1 First set of Experiments

In this first set of experiments, we consider a network that consists of 1 AP and 5 STAs. The AP generates 5 parallel UDP sessions of load 5 Mbps, one with each STA. Each experiment is repeated five times. In each experiment, we sequentially activate the features of our mechanism, starting from the default algorithm of Mad-WiFi and ending with the full activation of the metric as this is stated in equation (5). The aim of this experimental set is to estimate how different features of our mechanism cope with the uncontrolled external interference that exists in the outdoor testbed. The aggregate network throughput, according to the approach followed in each discrete experiment, is presented in Fig. 4(a).

In experiment 1, we use the default Mad-WiFi algorithm. The AP chooses channel 2, since this is the one that offers the lowest RSSI values, based solely on its measurements. By inspecting the scanning results offline, we confirmed that APs operating on channel 2 provided the lowest RSS in the reception of their *Beacon* frames. As seen in Fig. 4(a), the performance of the default algorithm is not efficient at all, and this indicates that additional factors must be considered too.

In the second experiment, we use a simplified version of our algorithm that enables the AP to take into account the degree of overlapping in signal power among adjacent channels as well. The result of this approach is that the AP manages to detect channel overlapping and selects channel 7 which leads to significant throughput increase, as depicted in Fig. 4(a).

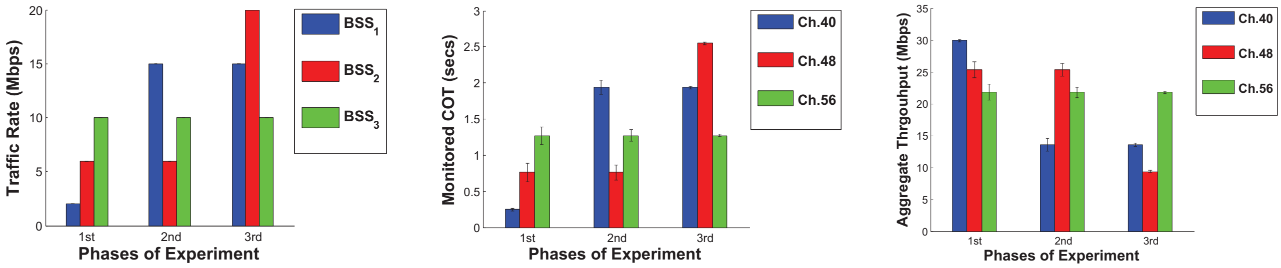
During the next experiment, the client-feedback feature of our mechanism is enabled. This approach involves the STAs in interference estimation as well. First, the AP combines the measurements of its 5 associated STAs and thus decides to operate on channel 10. As presented in Fig. 4(a), measurements collected by the limited number of 5 associated STAs do not lead to notable throughput increase.

In the fourth experiment, we utilize the feature of our implementation that enables STAs belonging to neighboring cells to exchange *Measurement Report* frames. We activate a new part of the network that consists of 2 BSSs, each of them having 1 AP and 5 STAs. Both of the new BSSs run the proposed DFS scheme and operate initially on channel 8. According to this approach, the AP utilizing measurements from a total of 15 neighboring STAs decides to operate on channel 8. Fig. 4(b) depicts the throughput performance measured by an offline execution of the same experiment for the same duration, where static channel assignments were forced. The proposed algorithm efficiently chooses the optimal channel (channel 8) that indeed gives the best performance among all available ones.

In the final experiment, we use only two APs without any associated STAs. The AP under consideration takes into account the information supported by its associated STAs and detects high RSSI values on channel 8, and thus it decides to evacuate it and switch to channel 4. This last channel transition results in throughput reduction. This is due to the fact that the two APs, capturing the medium for infinitesimally small duration just to transmit *Beacon* frames, practically do not affect COT on channel 8. As a result, the limited effect that they cause in channel congestion should have been properly considered. Moreover, in Fig. 4(c), we present results obtained from a node running in monitor mode capturing frames that were transmitted during the execution of this set of experiments. These results show that a clear relation between traffic conditions and throughput performance achieved on each channel. The conclusions drawn, based on the last results, clearly show that traffic estimation is an integral part of our DFS algorithm.

8.2 Second set of Experiments

The second set of experiments is designed to evaluate performance of networks operating on non-overlapping channels. For this purpose, we design a network setup that consists of 3 different BSSs operating on separate channels of the 5 GHz band, spaced 40 MHz apart. More specifically, each one of these 3 BSSs consists of one AP and one associated STA, where AP1 is assigned Ch. 40, AP2 channel 48 and AP3 Ch.56. These 3 BSSs are used to generate interference conditions of controlled traffic. We also use a fourth BSS that consists of one AP and 2 associated STAs, which runs a special version of our DFS algorithm that does not



(a) Traffic Rate in each BSS - Phases.

(b) COT on each channel- Phases.

(c) Throughput of BSS4 - Phases.

Figure 5: Second set of Experiments - Uplink Outdoor Experiments in the 5 GHz band.

consider channel overlapping. The aim of this experimental set is to show how our algorithm adapts to varying traffic conditions. Through driver level modifications, we limit the set of available channels to the three used by the interfering APs. This set is conducted in three phases, among which the traffic rate induced by each STA is varied, while it remains constant during each discrete step. Fig. 5(a) shows the traffic rate used by each STA in each discrete phase. Moreover, the average COT values, as reported by the corresponding APs in their *Beacon* frames among the phases of the experiment, are portrayed in Fig. 5(b).

During each phase, the AP of BSS4 manages to detect channel contention by calculating the COT value per channel. As a result, it decides to operate on channel 40 in the first phase, channel 48 in the second and channel 56 in the third phase. Fig. 5(c) presents the aggregate network throughput performance achieved by the fourth BSS as measured by an offline execution of the same experiment for the same duration, where static channel assignments were forced. It is clearly depicted that the proposed scheme results in channel selections that deliver the highest available throughput for all cases.

8.3 Third set of Experiments

In order to corroborate the results obtained through our previous experiments, we also have to conduct experiments with all the various features simultaneously enabled. The preceding experiments were restricted in a disjoint consideration of overlapping channels interference and traffic conditions. To take a step further, it is important to have an environment that ensures controlled conditions for both of the above factors. Thus, we run the next experiment in an RF-isolated environment that lies in the premises of University of Thessaly. This indoor testbed, which experiences zero external interference, consists of 6 laptops equipped with a single wireless interface that uses the Atheros AR5424 chipset.

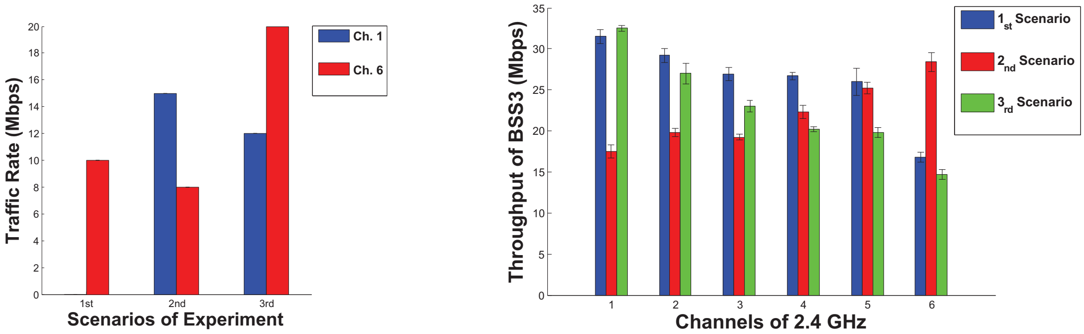
The experimental setup, designed under the restriction imported by the limited number of available nodes, is composed of 3 different BSSs, each one consisting of a single AP-STA pair. By modifying the driver, we have limited the set of available channels to the first 6 channels of the 2.4 GHz band, in order to simulate interfering scenarios in the entire range of 5 consecutive overlapping channels. The first 2 BSSs are assigned channels 1 and 6, while BSS3 is running the implemented DFS algorithm with the complete list of features simultaneously enabled. We consider only downlink scenarios, where the APs generate UDP traffic of certain traffic rate. In this experimental set, we present three representative scenarios, among the various topologies and configurations used. Fig. 6(a) presents the constant traffic rate, generated by the APs, during each discrete step. Note that due to the close spacing of nodes in the indoor testbed, we observed that there still exists interference, even for nodes operating on channels 1 and 6. Through proper TX_{power} reduction, we removed this effect caused due to the famous *"Near-Far"* phenomenon in order to provide for a proper measurement setup.

During the first scenario, BSS1 operating on channel 6 generates a traffic flow at the rate of 10 Mbps that causes interference to channels 2 till 6. BSS3, following our DFS approach, selects to operate on channel 1, as this is the only one, non-affected by the interfering link. The original driver performs the scanning in the order 1-11 and as a result selects channel 1 as it offers zero RSS values. The throughput performance of BSS3, measured in each scenario by an offline execution of the same experiment, is presented in Fig. 6(b). In the second scenario, two interfering links are active at the same time, the first on channel 6 transmitting at the rate of 8 Mbps, while the second uses channel 1 to transmit at the rate of 15 Mbps. In this case, our algorithm selects channel 6, avoiding the channel used by the link of higher load, as well as the intermediate channels (2-5), which influenced by both of the interfering links offer low performance. Finally, in the third scenario, BSS1 uses channel 1 to transmit at the rate of 12 Mbps, while BSS2 transmits on channel 6 at the rate of 20 Mbps. BSS3, using our mechanism, decides to operate on channel 1, while the original Mad-WiFi selects channel 2 in both scenarios 2 and 3, because it offers zero RSS. Clearly, our approach offers the highest throughput for all the scenarios.

A particular observation in this scenario is the fact that performance of the interfering link on channel 6, is significantly degraded when BSS3 starts transmitting, which leads to the particularly high throughput achieved. This phenomenon is related to the *"Capture Effect"*, which occurs in cases where frames received with high RSS suffer only few frame losses, as opposed to frames that are simultaneously received, but with relatively low RSS values. According to our experiments, certain topology and transmission power configurations lead the capture effect to either favor the link of BSS3 or the interfering links. These observations yield interesting insights regarding the impact of the *"Capture Effect"* on interference and motivate further investigation.

8.4 Overhead Consideration

We now discuss the overhead that the proposed scheme imports. It is known that the period at which the *Measurement Report* packet is broadcasted exerts crucial influence



(a) Traffic Rate of each BSS - Scenarios. (b) Throughput of BSS 3 - Scenarios.

Figure 6: Third set of Experiments - Downlink Indoor Experiments in the 2.4 GHz band.

on the system performance. In all our experiments, we used *Measurement Report* frames that are periodically transmitted every 30 secs. However, as frequency switching is not required to take place rather frequently, even higher values could have been used. Having conducted several experiments of varying interval values, we conclude that throughput reduction caused due to transmissions of *Measurement Report* frames is minimal, due to their relatively small size (51 bytes) and low transmission frequency.

Another factor that can severely affect performance is the frequency at which BG_{scan} procedures are performed. Nevertheless, through experimentation we observed that normal transmissions are prioritized over BG_{scan} procedures. In detail, scheduled BG_{scans} are canceled, when there are transmissions to be performed. This mechanism is not part of our work, but a characteristic provided by the Mad-WiFi driver. As a result, there is no overhead induced by the scanning procedure in the original driver, which is performed only when the node is idle. Due to this feature, highly congested APs do not manage to get their own measurements. Our protocol provides support to congested APs, with the modification of the original BG_{scan} to support scanning in batches of smaller size, so that their BG_{scans} are less frequently canceled and moreover by providing them with suitable measurements from neighboring STAs, through our feedback mechanism.

9. CONCLUSIONS AND FUTURE WORK

In this work, we proposed and implemented a DFS scheme that is based on innovative metrics for 802.11 WLANs. The proposed algorithm enables APs to collect measurements from neighboring STAs in an effort to make a better estimate of both channel as well as traffic conditions at different frequencies. The experimental results indicate significant improvement of user performance in realistic conditions. We plan to extend our work to the case where APs will perform load balancing to distribute the number of associated STAs equally. In addition, we intend to jointly consider the proposed scheme with the user association approach, proposed in our previous work [15].

10. ACKNOWLEDGEMENTS

The authors acknowledge the support of the European Commission through STREP project CONECT (FP7-257616).

11. REFERENCES

[1] ”Madwifi Driver”, http://madwifi-project.org/.

[2] *”Cisco Aironet”*, http://cisco.biz/.

[3] I. Katzela and M. Naghshineh. Channel Assignment Schemes for Cellular Mobile Telecommunication Systems: A Comprehensive Survey. 1996.

[4] E.G. Villegas, R.V. Ferro, and J.P. Aspas. Implementation of a Distributed Dynamic Channel Assignment Mechanism for IEEE 802.11 Networks. In *Proceedings of PIMRC*, 2005.

[5] B. Kauffmann, F. Baccelli, A. Chaintreau, V. Mhatre, K. Papagiannaki, and C. Diot. Measurement-Based Self Organization of Interfering 802.11 Wireless Access Networks. In *Proceedings of INFOCOM*, 2007.

[6] A. Mishra, V. Brik, S. Banerjee, A. Srinivasan, and W. Arbaugh. A client-driven approach for channel management in wireless LANs. In *Proceedings of Infocom*, 2006.

[7] A. Mishra, V. Shrivastava, D. Agarwal, S. Banerjee, and S. Ganguly. Distributed channel management in uncoordinated wireless environments. In *Proceedings of ACM Mobicom*, 2006.

[8] E. Rozner, Y. Mehta, A. Akella, and Lili Qiu. Traffic-Aware Channel Assignment in Enterprise Wireless LANs. In *Proceedings of ICNP*, 2007.

[9] G. Athanasiou, I. Broustis, T. Korakis, and L. Tassiulas. LAC: Load-Aware Channel selection in 802.11 WLANs. In *Proceedings of PIMRC*, 2008.

[10] A. Raniwala and Tzi cker Chiueh. Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network. In *Proceedings of INFOCOM*, 2005.

[11] G. Athanasiou, I. Broustis, T. Korakis, and L. Tassiulas. Routing-Aware Channel Selection in Multi-Radio Mesh Networks. In *Proceedings of ICC*, 2009.

[12] A. Mishra, V. Shrivastava, S. Banerjee, and W. Arbaugh. Partially overlapped channels not considered harmful. In *Proceedings of SIGMETRICS*, 2006.

[13] IEEE 802.11-2003. Amendment 5: Spectrum and Transmit Power Management Extentions in the 5 ghz Band in Europe, 2003.

[14] *”NITOS Wireless Testbed”*, http://nitlab.inf.uth.gr.

[15] S. Keranidis, T. Korakis, I. Koutsopoulos, and L. Tassiulas. Contention and Traffic Load-aware Association in IEEE 802.11 WLANs: Algorithms and Implementation. In *Proceedings of WinMee*, 2011.