UNIVERSITY OF THESSALY, GREECE

DESIGN AND IMPLEMENTATION OF RESOURCE ALLOCATION ALGORITHMS IN WIRELESS NETWORKS

Doctor of Philosophy
in
Electrical & Computer Engineering
by

STRATOS KERANIDIS

July 2014

Dissertation Committee:

Prof. Leandros Tassiulas
Assistant Prof. Iordanis Koutsopoulos
Associate Prof. Ioannis Katsavounidis

Stratos Keranidis: *Design and Implementation of Resource Allocation Algorithms in Wireless Networks,* Users, Spectrum, Energy, © June 2014

The Dissertation of Stratos Keranidis is approved by:

_____

_____

_____

Committee Chairperson

Dept. of Electrical & Computer Engineering,
University of Thessaly, Greece

Dedicated to my grandfather, Ntinos.

ABSTRACT

DESIGN AND IMPLEMENTATION OF RESOURCE ALLOCATION ALGORITHMS IN WIRELESS NETWORKS

by STRATOS KERANIDIS

DOCTOR OF PHILOSOPHY IN ELECTRICAL & COMPUTER ENGINEERING

UNIVERSITY OF THESSALY, GREECE

PROF. LEANDROS TASSIULAS CHAIRPERSON

Efficient management of available resources in wireless networks is directly related with the optimisation of network performance and as a result is actively investigated by the research community in the field of wireless networks. In this thesis, we study resource allocation algorithms in wireless networks, considering the resources of network users, the utilised spectrum and the amount of energy that is being consumed in different phase of network operation. The common research approach that has been followed across all the development phases of our work, is based on the design and implementation of the proposed algorithms in real wireless devices , in order to assess performance under realistic environments and provide for direct comparison with existing standards.

We start by studying user association algorithms, which assign at assigning network terminals to the available access points and more specifically focused on wireless networks that are compatible with the IEEE 802.11 protocol, commonly known as Wi-Fi. The existing protocol defines a rather simple association policy, based on which each terminal is associated with the access point from which it has received the strongest signal during the scanning process. This rather simplistic association policy has been shown in the literature that is not able to result in assignments that maximize throughput performance, as it does not take into account factors that affect the performance of 802.11 compatible networks. We briefly mention the contention between neighboring nodes for use of the wireless medium, the number of already already associated with each access point terminals along with the network load they result in, and the interference that is generated by transmissions taking place in other networks. Our work has resulted in the design of a novel distributed algorithm that jointly considers the above factors and leads to assignments that maximize the throughput of each individual ter-

minal. Comparative study between the implemented mechanism and the 802.11 based approach, in realistic experiments and under varying network load has shown significant throughput improvement (65 %). We also observed the distribution of throughput between terminals to be almost equal contrary to the performance of 802.11, which benefits only a subset of terminals that are located close to under-loaded access points.

As a next step, we extended the proposed mechanism by taking into account not only the individual user performance, but the overall network throughput as well. In order to evaluate the performance of the proposed algorithms, we conducted several sets of experiments and experimentally validated the superiority of the proposed algorithms contrary to the 802.11 based approach. Finally, we studied user association mechanisms the consider performance under combined topologies, consisting of both wireless and wired parts. Study of implemented algorithms was conducted in combined wired and wireless network infrastructure and showed the benefits that can be offered by association mechanisms that take into account metrics able to characterize end user performance in combined topologies .

In the context of assigning the spectrum of operation in wireless networks, we studied mechanisms that select the operational frequency in order to avoid congested bands and bands that experience strong levels of interference from neighboring transmitters. According to the current standard, configuration of operational frequency is executed at the access points by selecting the frequency that provides the minimum received power from neighboring networks. Existing commercial implementations that are based on the standard policy statically configure the frequency of operation, once during the device startup and come in contrast with the dynamic nature of the wireless medium. Aiming at improving the overall throughput of wireless networks, we proposed a novel mechanism that considers measurements collected at the acceess point along with measurements that have been gathered by adjacent stations. The proposed approach avoids frequency configurations that benefit only the local environment of the Base Station, as the operating frequency is selected based on combination of measurements that have been collected by all network nodes. An important advantage of the proposed scheme is also the ability of classifying the levels of network traffic in real time, thus resulting in efficient avoidance of congested frequencies. Moreover, since the proposed mechanism is applied dynamically, any changes in the wireless environment are immediately identified and consequently result in reconfiguration of the operating frequency. Having implemented the proposed mechanism in real wireless devices, we conducted extensive realistic experiments that proved the superiority of the proposed scheme compared to the standard policy, in terms of overall network throughput.

Next, we proceeded by extensively studying the influence of interference effects on the throughput of wireless networks and by designing mechanisms able to detect each type of interference. We also studied the influence of interference phenomena in the repeatability of wireless experiments. Based on the obtained results, we developed mechanisms able to characterize the stability of experimental conditions and consequently the validity of experimental results. The ultimate goal of these activities has been the development of the AGILE spectrum adaptation system that is able to adapt to the prevailing conditions under varying types of interference, in uncoordinated 802.11 WLANs. The key novelty of our approach is that identification of under-utilised spectrum fragments was able to take advantage of hardware inherent in the PHY-layer of standard OFDM receiver circuits. AGILE satisfied all the design requirements that uncoordinated network deployments specify - distributed operation, interoperability with existing standards, simple implementation and minimal protocol overhead. Through the implementation of our approach on commercial 802.11n chipsets and its detailed experimental evaluation, we showcased its feasibility and moreover quantified the obtained throughput performance and energy efficiency benefits. To the best of our knowledge, AGILE system was the first to enable dynamic spectrum adaptation in uncoordinated WLAN deployments and provide for direct implementation on off-the-self equipment.

The next part of our work is related with the energy efficient operation of wireless communication networks, which may result in significant waste of energy, both due to the rapid increase in carried data traffic as well as due to their inefficient in terms of energy mode of operation. Researchers working on designing energy efficient protocols are traditionally based on the use of models to estimate the amount of consumed energy. However, due to the proved inability of models to evaluate the energy consumption of complex networks, the development of research tools able to monitor the consumption of wireless devices accurately and in real time becomes a necessity. Having extensively studied the related literature, we identified a large gap in the available experimental infrastructure that can support the research community. Towards filling the identified gap, we developed a prototype electronic device, of particularly low cost, which enables detailed analysis of the energy consumed by distributed wireless devices in real time. As our next step, we integrated the initial version of the developed device into an operational framework ready to be incorporated into networking testbeds. The upgraded version of the device has been used towards evaluating the consumption of wireless communication networks under realistic scenarios and quantified the impact of exogenous factors (eg. interference, contention for channel use) on energy consumption, thus experimentally validating the inability of applying models in complex topologies. The developed platform has also been used to compare the energy efficiency of the 802.11 protocol with the upgraded version of 802.11n, which study showed that the latest

protocol version is able to offer reduction of consumption per unit of transmitted information (bit), in the order of 75 % . The detailed findings of the above study provided guidelines for the design of energy efficient wireless protocols.

Τίτλος Διδακτορικής Διατριβής:
Σχεδιασμός και Υλοποίηση Αλγορίθμων ανάθεσης πόρων σε ασύρματα δίκτυα

Κερανίδης Ευστράτιος

Περίληψη Διδακτορικής Διατριβής:

Η αποδοτική διαχείριση των διαθέσιμων πόρων σε ασύρματα δίκτυα επικοινωνιών είναι άμεσα συνδεδεμένη με την βελτιστοποίηση της απόδοσης τους και ως αποτέλεσμα έχει καταστεί ένα βασικό πεδίο έρευνας στον τομέα των ασυρμάτων επικοινωνιών. Στην παρούσα διδακτορική διατριβή έχουν μελετηθεί αλγόριθμοι ανάθεσης πόρων σε ασύρματα δίκτυα επικοινωνιών, όπου ως πόρους έχουμε θεωρήσει τους χρήστες του εκάστοτε δικτύου, το φάσμα εκπομπής του, αλλά και την ενέργεια που καταναλώνεται σε διάφορες φάσεις της λειτουργίας του. Η κοινή ερευνητική προσέγγιση που ακολουθήθηκε σε όλες τις φάσεις εξέλιξης της δουλειάς μας, έχει βασιστεί στο σχεδιασμό και την υλοποίηση των προτεινόμενων αλγορίθμων σε πραγματικές ασύρματες συσκευές, με σκοπό την εκτίμηση απόδοσης σε πραγματικά περιβάλλοντα και σε σύγκριση με τα ισχύοντα πρωτόκολλα λειτουργίας.

Αρχικά μελετήσαμε αλγορίθμους ανάθεσης τερματικών-χρηστών στους διαθέσιμους Σταθμούς Βάσης και εστιάσαμε σε ασύρματα δίκτυα συμβατά με το πρωτόκολλο IEEE 802.11, το κοινώς γνωστό ως Wi-Fi. Σύμφωνα με το ισχύον πρωτόκολλο, κάθε τερματικό πρέπει πρώτα να συνδεθεί σε έναν Σταθμό Βάσης, προτού να αρχίσει η μετάδοση δεδομένων προς άλλους κόμβους του δικτύου. Το πρότυπο IEEE 802.11 καθορίζει μία πολύ απλοϊκή πολιτική συσχέτισης τερματικών σε Σταθμούς Βάσης, επιλέγοντας προς σύνδεση το σταθμό από τον οποίο έχει ληφθεί το ισχυρότερο σήμα κατά τη διαδικασία σάρωσης προς ανακάλυψη των διαθέσιμων επιλογών σύνδεσης. Η απλοϊκή αυτή πολιτική συσχέτισης έχει αποδειχθεί στη σχετική βιβλιογραφία, πως δεν μπορεί να οδηγήσει σε αναθέσεις που μεγιστοποιούν την ρυθμοαπόδοση (throughput), καθώς δεν λαμβάνει υπ' όψιν παράγοντες που επηρεάζουν την απόδοση των 802.11 συμβατών δικτύων. Συνοπτικά αναφέρουμε τον ανταγωνισμό προς χρήση του ασύρματου μέσου μεταξύ γειτονικών κόμβων, το πλήθος ήδη συνδεδεμένων τερματικών στον εκάστοτε Σταθμό Βάσης και τον φόρτο που εισάγουν, αλλά και την παρεμβολή που δημιουργείται από μεταδόσεις άλλων δικτύων. Αποτέλεσμα της δουλειάς μας υπήρξε ο σχεδιασμός ενός πρωτότυπου κατανεμημένου αλγορίθμου, ο οποίος παρατηρεί συνδυασμένα την εξέλιξη των παραπάνω παραγόντων και οδηγεί σε αναθέσεις που μεγιστοποιούν τη ρυθμοαπόδοση του εκάστοτε τερματικού. Συγκριτική μελέτη της απόδοσης του υλοποιημένου μηχανισμού σε σχέση με την πολιτική που εφαρμόζει το 802.11, σε ρεαλιστικά πειράματα υπό μεταβαλλόμενο δικτυακό φόρτο, έδειξε σημαντική βελτίωση (κατά 65%) της ρυθμοαπόδοσης. Επίσης παρατηρήσαμε πως η κατανομή της ρυθμοαπόδοσης μεταξύ των τερματικών του δικτύου υπήρξε σχεδόν ίση, σε αντίθεση με την απόδοση του 802.11, το οποίο ευνοεί μόνο το υποσύνολο τερματικών τα οποία βρίσκονται κοντά σε Σταθμούς Βάσης που αντιμετωπίζουν χαμηλό φόρτο κίνησης.

Στη συνέχεια επεκτείναμε τον προτεινόμενο από εμάς μηχανισμό, ώστε να λαμβάνει υπ' όψιν του όχι μόνο την απόδοση του εκάστοτε τερματικού, αλλά και τη συνολική ρυθμοαπόδοση ολόκληρου του δικτύου. Σειρές πειραμάτων εκτελέστηκαν για την συγκριτική αξιολόγηση της απόδοσης των δύο προτεινόμενων αλγορίθμων υπό διαφορετικές τοπολογίες και σενάρια και έδειξαν την υπεροχή τους σε σχέση με τον προτεινόμενο από το 802.11 μηχανισμό. Τέλος, ασχοληθήκαμε με τη μελέτη μηχανισμών συσχέτισης τερματικών με Σταθμούς Βάσης, οι οποίοι εξετάζουν την

απόδοση συνδυασμένων τοπολογιών που αποτελούνται από ασύρματα, αλλά και ενσύρματα τμήματα. Μελέτη των υλοποιημένων αλγορίθμων διεξήχθει συνδυασμένα σε ενσύρματες και ασύρματες πειραματικές υποδομές (testbeds) και έδειξε τα οφέλη που μπορούν να προσφέρουν αλγόριθμοι συσχέτισης τερματικών σε Σταθμούς Βάσης, οι οποίοι λαμβάνουν υπ' όψιν μετρικές που χαρακτηρίζουν την απόδοση του τελικού χρήστη, όπως αυτή διαμορφώνεται σε συνδυασμένες τοπολογίες.

Στα πλαίσια της μελέτης ανάθεσης φάσματος εκπομπής σε ασύρματα δίκτυα, ασχοληθήκαμε με μηχανισμούς επιλογής βέλτιστης συχνότητας λειτουργίας, οι οποίοι στοχεύουν στην αποφυγή συμφορημένων συχνοτήτων, αλλά και συχνοτήτων που αντιμετωπίζουν ισχυρά επίπεδα παρεμβολής από γειτονικούς μεταδότες. Σύμφωνα με το ισχύον πρωτόκολλο 802.11, η επιλογή συχνότητας λειτουργίας αποφασίζεται τοπικά στον εκάστοτε Σταθμό Βάσης, επιλέγοντας τη συχνότητα που προσφέρει τα ελάχιστα επίπεδα ισχύος λήψης από γειτονικά δίκτυα. Υπάρχουσες εμπορικές υλοποιήσεις εφαρμόζουν την πολιτική που καθορίζει το πρότυπο και επιλέγουν στατικά τη συχνότητα λειτουργίας, μία φορά κατά την έναρξη λειτουργίας της συσκευής, ερχόμενοι σε αντίθεση με τη δυναμική φύση του ασύρματου μέσου. Στοχεύοντας στη βελτίωση της συνολικής ρυθμοαπόδοσης των ασυρμάτων δικτύων, προτείναμε έναν πρωτότυπο μηχανισμό, ο οποίος συνδυάζει μετρήσεις που έλαβε ο Σταθμός Βάσης με μετρήσεις που λήφθησαν από γειτονικούς σταθμούς. Καθώς πλέον η συχνότητα λειτουργίας έχει επιλεγεί συνδυασμένα από όλους τους κόμβους του εκάστοτε δικτύου, αποφεύγονται επιλογές που ευνοούν αποκλειστικά το τοπικό περιβάλλον του Σταθμού Βάσης. Σημαντικό πλεονέκτημα του προτεινόμενου σχήματος αποτελεί επίσης η δυνατότητα χαρακτηρισμού των επιπέδων δικτυακής κίνησης σε πραγματικό χρόνο, η οποία οδηγεί στην αποτελεσματική αποφυγή συμφορημένων συχνοτήτων. Επίσης καθώς ο προτεινόμενος μηχανισμός εφαρμόζεται δυναμικά, τυχόν μεταβολές στο ασύρματο περιβάλλον εντοπίζονται άμεσα, με συνέπεια την άμεση μεταβολή της συχνότητας λειτουργίας. Έχοντας υλοποιήσει των προτεινόμενο μηχανισμό σε πραγματικές ασύρματες συσκευές, διεξαγάγαμε εκτενή ρεαλιστικά πειράματα, τα οποία έδειξαν την υπεροχή του προτεινόμενου σχήματος σε σχέση με τον ισχύοντα μηχανισμό, όσον αφορά την επιτεύξιμη ρυθμαπόδοση του συνολικού δικτύου.

Επόμενο βήμα στα πλαίσια της μελέτης μας, υπήρξε η εκτενής μελέτη της επιρροής φαινομένων παρεμβολής στη ρυθμοαπόδοση ασυρμάτων δικτύων, καθώς και ο σχεδιασμός μηχανισμών ανίχνευσης του εκάστοτε τύπου παρεμβολής. Επίσης μελετήθηκε η επιρροή φαινομένων παρεμβολής στην επαναλειψημότητα που εμφανίζει η εκτέλεση ασύρματων πειραμάτων. Με βάση τα αποτελέσματα που προέκυψαν αναπτύξαμε μηχανισμούς που παρέχουν τη δυνατότητα χαρακτηρισμού της σταθερότητας των πειραματικών συνθηκών και κατά συνέπεια της εγκυρότητας των πειραματικών αποτελεσμάτων. Απώτερος στόχος των παραπάνω δραστηριοτήτων υπήρξε η ανάπτυξη του συστήματος AGILE που μπορεί να προσαρμόζει δυναμικά το φάσμα εκπομπής σε σχέση με τις ισχύουσες συνθήκες παρεμβολής, σε μη διαχειριζόμενα 802.11 ασύρματα δίκτυα. Η κύρια καινοτομία της προσέγγισης μας, έγκειται στο μηχανισμό αναγνώρισης υπό-κατηλειμένων τμημάτων του φάσματος, ο οποίος χρησιμοποιεί πληροφορία που προκύπτει από κατάληλο κύκλωμα του Φυσικού επίπεδο συμβατικών OFDM συσκευών. Το σύστημα AGILE ικανοποιεί όλες τις σχεδιαστικές απαιτήσεις που εισάγουν τα μη διαχειριζόμενα δίκτυα, όπως κατανεμημένη λειτουργία, διαλειτουργικότητα με τα ισχύοντα πρωτόκολλα, απλή υλοποίηση και ελάχιστο φόρτο πρωτοκόλλου. Μέσω της υλοποίησης της προσέγγισης μας σε εμπορικές 802.11n κάρτες και της λεπτομερούς πειραματικής αξιολόγησης, παρουσιάσαμε την εφικτότητα της προσέγγισης μας, ενώ επιπρόσθετα ποσοτικοποιήσαμε τα οφέλη ρθυμοαπόδοσης και ενεργειακής απόδοσης που είναι σε θέση να προσφέρει. Με βάση τη γνώση μας για σχετικές δουλειές του χώρου, αναφέρουμε πως το σύστημα AGILE υπήρξε το πρώτο που κατέστισε ικανή τη

δυναμική προσαρμογή φάσματος σε μη διαχειριζόμενα 802.11 ασύρματα δίκτυα και την υλοποίηση της σε εμπορικές συσκευές.

Επόμενο τομέα ενασχόλησης μας, αποτέλεσε η μελέτη της ενεργειακής αποδοτικότητας ασυρμάτων δικτύων επικοινωνιών, τα οποία λόγω της ραγδαίας αύξησης της κίνησης δεδομένων που μεταφέρουν, αλλά και του ενεργειακά μη αποδοτικού τρόπου λειτουργίας τους, οδηγούν στην κατανάλωση τεράστιων ποσοτήτων ενέργειας. Στα πλαίσια σχεδιασμού ενεργειακά αποδοτικών πρωτοκόλλων, οι ερευνητές του χώρου, παραδοσιακά στηρίζονται στη χρήση μοντέλων εκτίμησης της καταναλισκόμενης ενέργειας. Καθώς όμως η σχετική βιβλιογραφία έχει αποδείξει πως η χρήση μοντέλων είναι ακατάλληλη για τη μελέτη κατανάλωσης σύνθετων δικτύων, καθίσταται αναγκαία η χρήση ερευνητικών εργαλείων που μπορούν να παρακολουθούν την κατανάλωση ασυρμάτων συσκευών με ακρίβεια και σε πραγματικό χρόνο. Μετά από εκτενή μελέτη της σχετικής βιβλιογραφίας, εντοπίσαμε ένα μεγάλο κενό στις διαθέσιμες πειραματικές υποδομές που μπορούν να υποστηρίξουν τους ερευνητές του χώρου. Στοχεύοντας στη δημιουργία κατάλληλων υποδομών, αναπτύξαμε μια πρωτότυπη ηλεκτρονική διάταξη, ιδιαίτερα χαμηλού κόστους, η οποία παρέχει τη δυνατότητα λεπτομερούς ανάλυσης της ενέργειας που καταναλώνεται από ασύρματες συσκευές, κατανεμημένα και σε πραγματικό χρόνο. Στη συνέχεια προχωρήσαμε ολοκληρώνοντας την αρχική έκδοση της διάταξης σε ένα ενιαίο εργαλείο έτοιμο προς ενσωμάτωση σε δίκτυα πειρματικών υποδομών. Η αναβαθμισμένη έκδοση της συσκευής χρησιμοποιήθηκε κατά τη μέτρηση κατανάλωσης ασυρμάτων δικτύων επικοινωνίων υπό ρεαλιστικά σενάρια και ποσοτικοποίησε τον αντίκτυπο εξωγενών παραγόντων (π.χ. παρεμβολή, ανταγωνισμός χρήσης καναλιού) στην συνολικά καταναλισκόμενη ενέργεια, αποδεικνύοντας πειραματικά την αδυναμία εφαρμογής μοντέλων σε σύνθετες τοπολογίες. Η πλατφόρμα που αναπτύξαμε χρησιμοποιήθηκε επίσης για τη σύγκριση της ενεργειακής αποδοτικότητας του πρωτοκόλλου 802.11 με την αναβαθμισμένη έκδοση του 802.11n, η οποία μελέτη έδειξε πως το νεότερο πρωτόκολλο μπορεί να επιφέρει μείωση της κατανάλωσης ανά μονάδα μεταδιδόμενης πληροφορίας (bit), της τάξεως του 75%. Τα λεπτομερή ευρήματα της παραπάνω μελέτης έδωσαν κατευθυντήριες γραμμές για το σχεδιασμό ενεργειακά αποδοτικών ασυρμάτων πρωτοκόλλων.

PUBLICATIONS

The results, the ideas and figures are included in the following publications:

Journals and Magazines

[J.1.] S. Keranidis, T. Korakis, I. Koutsopoulos and L. Tassiulas, *"**Bringing Contention and Traffic Load Awareness into Access Point Selection in CSMA-based wireless networks**", under submission to Elsevier Computer Networks.*

[J.2.] S. Keranidis, G. Kazdaridis, V. Passas, T. Korakis, I. Koutsopoulos and L. Tassiulas *"**NITOS Energy Monitoring Framework: Real time Power Monitoring in Experimental Wireless Network Deployments**", in ACM Mobile Computing and Communications Review (MC2R), Special issue, January 2014.*

[J.3.] S. Keranidis, D. Giatsios, T. Korakis, I. Koutsopoulos, L. Tassiulas, T. Rakotoarivelo, M. Ott and T. Parmentelat, *"**Experimentation on end-to-end performance aware algorithms in the federated environment of the heterogeneous PlanetLab and NITOS testbeds**", in Elsevier Computer Networks, Special issue on Future Internet Testbeds, 2013.*

[J.4.] K. Choumas, S. Keranidis, T. Korakis, I. Koutsopoulos, L. Tassiulas, F. Juraschek, M. Gunes, E. Baccelli, P. Misiorek, A. Szwabe, T. Salonidis and H. Lundgren *"**Optimization driven Multi-Hop Network Design and Experimentation: The Approach of the FP7 Project OPNEX**", in IEEE Communications Magazine, Radio Communication Series, June 2012.*

Conferences

[C.1.] S. Keranidis, K. Chounos, T. Korakis, I. Koutsopoulos and L. Tassiulas, *"**Enabling AGILE Spectrum Adaptation in uncoordinated WLANs and commercial 802.11 devices**", submitted to ACM HotNets 2014.*

[C.2.] S. Keranidis, V. Passas, K. Chounos, W. Liu, T. Korakis, I. Koutsopoulos, I. Moerman and L. Tassiulas, *"**Online Assessment of Sensing Performance in Experimental Spectrum Sensing Platforms**", to be presented in ACM WinTech 2014, Mobicom 2014, Maui, Hawaii, September 2014.*

[C.3.] S. Keranidis, G. Kazdaridis, N. Makris, T. Korakis, I. Koutsopoulos and L. Tassiulas, ***"Experimental Evaluation and Comparative Study on Energy Efficiency of the Evolving IEEE 802.11 Standards"***, *in the Proceedings of ACM e-Energy 2014, Cambridge, UK, June 2014.*

[C.4.] S. Keranidis, G. Kazdaridis, V. Passas, T. Korakis, I. Koutsopoulos and L. Tassiulas, ***"Online Energy Consumption Monitoring of Wireless Testbed Infrastructure through the NITOS EMF Framework"***, ***Best Paper Award***, *in the Proceedings of ACM WinTech 2013, MobiCom 2013, Miami, Florida, September 2013.*

[C.5.] S. Keranidis, W. Liu, M. Mehari, P. Becue, S. Bouckaert, I. Moerman, T. Korakis, I. Koutsopoulos and L. Tassiulas, ***"CONCRETE: A benchmarking framework to CONtrol and Classify REpeatable Testbed Experiments"***, *in FIRE Engineering Workshop 2012, Gent, Belgium, November 2012.*

[C.6.] W. Liu, S. Keranidis, M. Mehari, J. V. Gerwen, S. Bouckaert, O. Yaron and I. Moerman, ***"Various Detection Techniques and Platforms for Monitoring Interference Conditions in Wireless Testbeds"***, *in LNCS of the Workshop in Measurement and Measurement Tools 2012, Aalborg, Denmark, May 2012.*

[C.7.] S. Keranidis, D. Giatsios, T. Korakis, I. Koutsopoulos, L. Tassiulas, T. Rakotoarivelo and T. Parmentelat, ***"Experimentation in Heterogeneous European Testbeds through the Onelab Facility: The case of PlanetLab federation with the wireless NITOS Testbed"***, *in the Proceedings of TridentCom 2012, Thessaloniki, Greece, June 2012.*

[C.8.] G. Kazdaridis, S. Keranidis, A. Fiamegkos, T. Korakis, I. Koutsopoulos and L. Tassiulas. ***"Novel Metrics and Experimentation Insights for Dynamic Frequency Selection in Wireless LANs"***, *in the Proceedings of ACM WiNTECH 2011, MobiCom 2011, Las Vegas, Nevada, USA, September 2011.*

[C.9.] S. Keranidis, T. Korakis, I. Koutsopoulos and L. Tassiulas, ***"Contention and Traffic Load-aware Association in IEEE 802.11 WLANs: Algorithms and Implementation"***, *in the Proceedings of WinMee 2011, WiOpt 2011, Princeton NJ, USA, May 2011.*

Demonstrations

[D.1.] S. Keranidis, K. Chounos, T. Korakis, I. Koutsopoulos and L. Tassiulas, ***"Enabling AGILE Spectrum Adaptation in commercial 802.11 WLAN deployments"***, *to be presented in ACM Mobicom 2014, Maui, Hawaii, September 2014.*

[D.2.] S. Keranidis, G. Kazdaridis, V. Passas, G. Igoumenos, T. Korakis, I. Koutsopoulos and L. Tassiulas, ***"NITOS Mobile Monitoring Solution: Realistic Energy Consumption Profiling of Mobile Devices"***, *in the Proceedings of ACM e-Energy 2014, Oxford, UK, June 2014.*

[D.3.] S. Keranidis, G. Kazdaridis, V. Passas, T. Korakis, I. Koutsopoulos and L. Tassiulas, ***"Online Energy Consumption Monitoring of Wireless Testbed Infrastructure through the NITOS EMF Framework"***, *in the Proceedings of ACM WiNTECH 2013, Mobicom 2013, Miami, Florida, September 2013.*

[D.4.] V. Passas, K. Chounos, S. Keranidis, W. Liu, L. Hollevoet, T. Korakis, I. Koutsopoulos, I. Moerman and L. Tassiulas, ***"Online Evaluation of Sensing Characteristics for Radio Platforms in the CREW Federated Testbed"***, *in the Proceedings of ACM Mobicom 2013, Miami, Florida, September 2013.*

[D.5.] V. Passas, S. Keranidis, T. Korakis, I. Koutsopoulos and L. Tassiulas, ***"An Experimental Framework for Channel Sensing through USRP/GNU Radios"***, *in the Proceedings of TridentCom 2012, Thessaloniki, Greece, June 2012.*

[D.6.] G. Kazdaridis, S. Keranidis, H. Niavis, T. Korakis, I. Koutsopoulos and L. Tassiulas, ***"An Integrated Chassis Manager Card Platform featuring multiple sensor modules"***, ***Best Demo Award***, *in the Proceedings of TridentCom 2012, Thessaloniki, Greece, June 2012.*

[D.7.] N. Makris, S. Keranidis, D. Giatsios, T. Korakis, I. Koutsopoulos and L. Tassiulas, ***"Cross-testbed experimentation using the Planetlab-NITOS federation"***, *in the Proceedings of TridentCom 2012, Thessaloniki, Greece, June 2012.*

[D.8.] G. Kazdaridis, S. Keranidis, A. Fiamegkos, T. Korakis, I. Koutsopoulos and L. Tassiulas, ***"Dynamic Frequency Selection through Collaborative Reporting in WLANs"***, *in the Proceedings of ACM MobiCom 2011, Las Vegas, Nevada, USA, September 2011.*

In addition, our research efforts within the same period led to the following publications that are not directly related to this thesis:

Conferences

[C.10.] I. Haratcherev, M. Meo, Y. Zhang, Y. Hu, A. Conte, F. Idzikowski, L. Budzisz, F. Ganji, R. Bolla, O. Jaramildo Ortiz, R. Bruschi, A. Cianfrani, L. Chiaraviglio, A. Coiro, R. Gonzales, C. Guerrero, E. Ego, F. Matera, S. Keranidis, G. Kazdaridis and T. Korakis, ***"The TREND Experimental Activities on green Communication Networks"***, *in the Proceedings of TIWDC 2013, Genoa, Italy, September 2013.*

[C.11.] W. Liu, M. Mehari, S. Bouckaert, I. Moerman, S. Keranidis, P. V. Wesemael, S. Pollin, T. Korakis and I. Koutsopoulos, ***"A Benchmarking Framework for conducting Easy and Reliable Wireless Experiments"***, *in FIRE Engineering Workshop 2012, Gent, Belgium, November 2012.*

Demonstrations

[D.9.] S. Keranidis, I. Koutsopoulos, T. Korakis, and L. Tassiulas, ***"An Experimental Framework for Data Gathering and Analysis in Wireless Sensor Networks."***, *in the Proceedings of ACM WiNTECH 2011, MobiCom 2011, Las Vegas, Nevada, USA, September 2011.*

ACKNOWLEDGMENTS

This thesis represents the milestone of more than a decade of studies in the Department of Electrical & Computer Engineering of University of Thessaly in Greece. During this long and wonderful journey, a lot of people have supported my efforts and helped me find the way towards my PhD Degree.

First and foremost I wish to thank my advisors, *Prof. Leandros Tassiulas, Assistant Prof. Iordanis Koutsopoulos and Lecturer Thanasis Korakis*, for giving me the opportunity to work in such an inspiring research unit and more importantly for their guidance, motivation and provisioning of the funding required for my research. Particularly, I would like to thank Iordanis Koutsopoulos, for his patience and understanding especially during the first years of my PhD studies. At that time his belief in me provided the support required to overcome the initial problems encountered and keep working with passion.

I have no words to express my gratitude to Thanasis Korakis for mentoring my research day by day. He made me believe in my capabilities and encouraged me to keep up trying even when there was no evident progress. Thank to him, me and my lab-mates feel proud of being members of the Network Implementation Testbed Laboratory (NITLab). Parallel to being a great teacher for me in the lab, he has been my closest friend in Volos over the last 6 years. His continuous support and advise in all aspects of life have totally changed me as a person.

Especially, I would like to thank my friends and lab-mates in NITlab, for supporting me every day. I totally believe that great achievements require team work and it is always better when help comes from friends. First of all, I would like to thank *Apostolis Apostolaras* for supporting me during my first days in the lab and for his huge efforts in maintaining the testbed. Particular thanks go to *Giannis Kazdaridis* for constructively assisting my work in all our joint research and for enabling me to build envisioned structures in hardware. I would really like to thank the newer lab-members, whose passion for work really motivates us all and more specifically *Virgilios Passas, Kostas Chounos and Ilias Syrigos*. Moreover, I would like to thank the rest NITLab members and friends *Kostas Choumas, Nikos Makris, Harris Niavis, Giannis Igoumenos, Dimitris Giatsios, Donatos Stavropoulos, Pavlos Bassaras, Kostas Katsalis, Stavros Ioannidis, Hristos Zarafetas, Antonis Kalkanov and Aris Dadoukis*. Particularly, I want to thank Giannis Igoumenos for

making the last year of my studies a very joyful experience. I would also like to thank *Christina Madelou* for her daily support in administration related issues. To *Wei Liu* I want to express my sincere thanks for supporting my research, during my internship in Belgium and thence forward.

Last but really important, I am more than grateful to my mother *Katerina* and father *Dimitris* for supporting me everyday all over these years. Without their support and love, I would not have come this far. I would also like to thank my brother *Kostis* and sister *Despoina* for being next to me in our whole lives. Especially, I thank *Kostis* for sharing his everyday life with me during the first years of our studies in the same department and my friend Gkelte for being more like a family member to me.

Especially, I would like to thank my grandfather *Ntinos* for shaping many aspects of my character when I was really young. I wish that a small part of his huge technical skills has passed to me as well. He was the person always requesting details about my work and progress. As he recently passed away and thus is not able to join the presentation of my work, I dedicate my dissertation to him.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1

INTRODUCTION

1.1 MOTIVATION

Over the last decade, the IEEE 802.11 [1] wireless standard that is widely known as WiFi, has emerged as the most popular protocol in the wireless domain. The wide adoption of this standard by vendors of wireless devices offers high interoperability, which in combination with the provided ease of use and low deployment cost have resulted in its unprecedented market and everyday life penetration. This tremendous growth has been highly impacted by the integration of the 802.11 standard in several types of mobile devices, including laptops, smartphones, tablet computers and home entertainment devices, such as electronic gaming consoles and Smart TVs. Recent studies [2] have shown that by 2015 the amount of devices with built-in WLAN capability will nearly double from the 2011 level, as illustrated in Fig. 1.

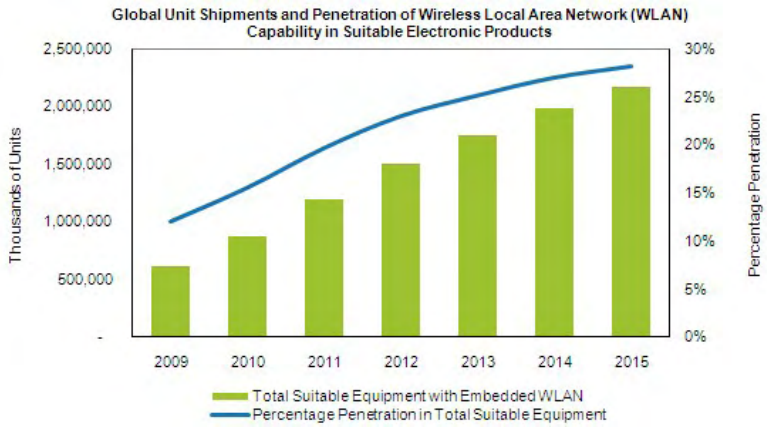


Figure 1: Penetration of Wi-Fi in electronic products

Especially in densely populated urban areas, a huge amount of wireless devices may simultaneously seek access to the common medium. In addition to above, the uncoordinated management of WLANs, in accordance with the limited width of the available unlicensed spectrum, constitute the wireless medium highly congested in such areas. In this unique context, efficient management of the available resources needs to be applied, in order to improve the performance experienced by end-users. Thus, resource allocation in wireless networks has become a major concern that is actively pursued by the global research community.

Depending on the actual point of view, the types of resources that constitute wireless networks may vary. Taking into account a high-level disaggregation of resources, we consider that in general wireless networks consist of the network access points and the users seeking access through them, the part of the spectrum that transmissions occupy and the amount of energy that network components consume. Following this approach, the research efforts of this thesis can be classified into three main threads, namely user association, spectrum management and energy efficiency.

1.1.1 *User Association*

Efficient distribution of users to the available network access points (APs) may determine to a large extent the overall network performance, as all frames exchanged in the resulting network, are either transmitted from the selected AP (downlink), or are destined to it (uplink). While the dense deployment of WLANs may provide several available association choices, it also constitutes the evaluation of potential associations a rather complex task. This comes mainly due to the fact that several factors, such as topology and channel characteristics, as well as interference and traffic conditions, have to be jointly considered.

As a result, global knowledge of network parameters and conditions should be considered towards driving efficient association decisions. However, due to the uncoordinated management and distributed operation of common WLAN deployments, association decisions can only based on local information that is collected by each individual user under association. Considering all the aforementioned factors, we conclude that user association in WLANs is a challenging problem that requires the integration of efficient decision making mechanisms with the distributed association mechanism that the 802.11 standard defines.

1.1.2 *Spectrum Management*

Devices compliant with the 802.11 standard are only able to transmit in the unlicensed 2.4 GHz and 5 GHz ISM bands. Especially, the popular 2.4 GHz band is currently overcrowded, as most APs are assigned operational frequencies within the band, while occupancy of the 5 GHz band is also constantly increasing. As a result, performance of 802.11 links is significantly affected, both due to medium congestion, as well as due to interference generated by neighboring transmitters. Taking also into account the prevailing traffic conditions, we clearly understand that dynamic reconfiguration

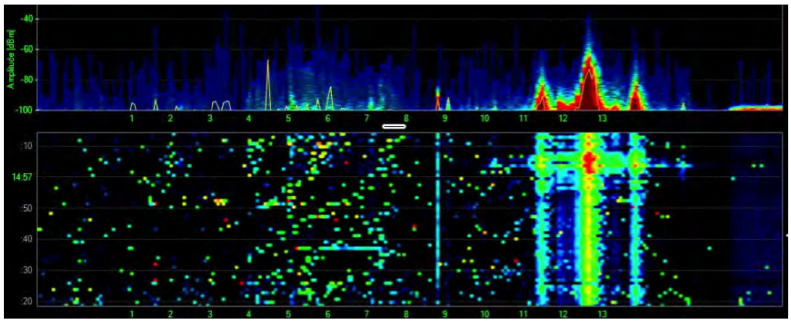


Figure 2: Indicative Spectrum occupancy over time and frequency in the 2.4 GHz band

of operational frequency needs to be applied, in order to successfully adapt to the dynamic nature of the wireless medium.

Another important characteristic of the 2.4 GHz band is that it is also home for other wireless protocols (e.g. Bluetooth, Zigbee) and a large range of RF devices (e.g. security cameras, baby monitors, microwave ovens). Such devices use variable channel bandwidth that spans up to several MHz and may either use a fixed central frequency or even perform frequency hopping, thus impacting a large part of the spectrum, as depicted in the indicative Spectrum snapshot in Fig. 2. As a result, WLANs should be able to detect transmissions generated by non-802.11 devices as well, in order to properly decide about the optimal part of the spectrum to operate on.

1.1.3 *Energy Efficiency*

Energy consumption minimization is considered as a major goal in numerous research fields, including wireless communications. While the information and technology ecosystem currently represents approximately 10% of the global electricity consumption [3], the network equipment and data centres are also characterized by the highest annual growth rate of 12% among other ICT infrastructure (Fig. 3). Moreover, it is important to remark that about 50% of the total energy consumption in the ICT sector is attributed to the wireless access part [4]. Therefore, network operators and service providers currently compete to optimize the energy efficiency of their access infrastructure in order to reduce both CO_2 emissions and operational costs [5].

On the other hand, the restricted battery autonomy of mobile devices, along with their high power consumption profile and the dramatic increase of exchanged mobile data (13x till 2017) bring evidence of the emerging problem to the end-user. Especially in the case of smartphone platforms, the energy greedy profile of the supported state-of-the-art wireless technologies may induce up to 50% of the total platform power consumption [6], under typical use case scenarios. The increased energy demands of

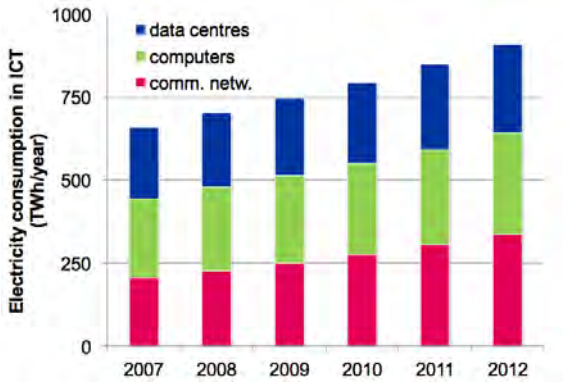


Figure 3: Evolution of ICT electricity consumption

such technologies cannot be successfully met, due to the limited energy capacity [7] that existing battery technologies are able to offer. The overall goal, towards alleviating this unique performance discrepancy, is to reduce energy consumption wherever possible. Thus, designing communication protocols and resource allocation mechanisms that take full advantage of the available energy is a goal of utmost importance.

1.2 SYNOPSIS

In this thesis, we study resource allocation algorithms in wireless networks, considering the resources of network users, the occupied spectrum and the amount of energy that is being consumed. The common research approach that has been followed across all the development phases of our work, is based on the design and implementation of the proposed algorithms in real wireless devices. Based on this approach, we were able to assess performance of the developed schemes under realistic environments and also to directly compare their efficiency contrary to existing standards.

In **Chapter 2**, we introduce innovative metrics for user association that are able to capture the effects of contention and interference in the neighborhood both on uplink and downlink. A key feature of the proposed scheme is that it manages to adapt to realistic traffic conditions. The proposed approach results in two discrete strategies that target the maximisation of individual node and aggregate network throughput performance accordingly. Compared with the standard RSSI-based approach, both of the implemented algorithms exhibit far better performance in terms of throughput and yielded fairness, as evaluated under realistic testbed experiments.

Next, in **Chapter 3**, we study user association mechanisms the consider performance under combined topologies, consisting of both wireless and wired parts. Study of implemented algorithms was conducted in combined wired and wireless network infrastruc-

ture and showed the benefits that can be offered by association mechanisms that take into account metrics able to characterize end user performance in combined topologies.

In **Chapter 4**, we develop a dynamic frequency selection algorithm that selects the operational frequency in order to avoid congested bands and bands that experience strong levels of interference from neighboring transmitters. 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. The proposed algorithm is implemented in commercial wireless cards 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.

In **Chapter 5 and Chapter 6**, we extensively study the influence of interference effects on the throughput of wireless networks. Particularly in Chapter 5, we design mechanisms able to detect each type of interference base on high-end spectrum sensing devices. We also study the influence of interference phenomena in the stability of experimental conditions and consequently the validity of experimental results. In Chapter 6, we focus on online assessment of spectrum occupancy with respect to sensing delay and energy consumption. We develop a sophisticated testbed experimentation tool that enables Cognitive Radio experimenters to evaluate the proposed spectrum sensing solutions, in terms of the aforementioned metrics.

In **Chapter 7**, we built on top of the experience on Spectrum Sensing that has been gained as an outcome of the research efforts conducted in Chapters 5 and 6. We develop a Spectrum Sensing mechanism that is able to operate on commercial 802.11 hardware and scans the spectrum on an Energy detection basis. Through this approach, we are able to detect the least congested part of the spectrum and properly configure both the central frequency and bandwidth of operation, in order to properly adapt to the prevailing interference and channel conditions. We also experimentally showcase the advantages of experiencing increased SNR, greater throughput under interfered conditions and decreased power consumption.

Next, in **Chapter 8**, we introduce a novel framework that is able to characterise the consumption of wireless testbed infrastructure in an online way. The proposed framework is built on a distributed network of low-cost, but highly accurate devices and is able to support online monitoring of energy expenditure, along with the experiment

execution. The overall system architecture provides highly accurate monitoring capabilities that aid towards energy performance assessment of realistic wireless scenarios.

Finally, in **Chapter 9**, we follow the power measurement methodology and use the hardware device that was introduced in Chapter 8, in order to experimentally evaluate the energy efficiency of the base 802.11 standard in comparison with the latest 802.11n version, under a wide range of settings. Based on in-depth interpretation of the collected results, we highlight the improvement in energy efficiency that the 802.11n version is able to provide. Our detailed findings can act as guidelines for researchers working on the design of energy efficient wireless protocols.

Part I

USER ASSOCIATION

2

CONTENTION AND TRAFFIC LOAD-AWARE USER ASSOCIATION IN IEEE 802.11 WLANS

2.1 INTRODUCTION

The IEEE 802.11 family of protocols is currently the standard for wireless lans (WLANs). Specifically for infrastructure 802.11 compliant networks, the standard specifies that each station (STA) has to first associate with a single access point (AP), before it can start transmitting data to other nodes in the network. Due to the rapidly increasing popularity of WLANs and the resulting high density deployments, quite often several available APs might exist, among which a STA can select one to associate with.

This association procedure consists of four phases. During the first phase, a STA has to discover the networks in its vicinity before it can join a Basic Service Set (BSS). This process is called *scanning* and can be either passive or active. In passive scanning, a STA scans all available channels and listens to information periodically broadcast by the APs in their *Beacon* frames. In active scanning, a STA tries to find the BSSs in its vicinity by transmitting *Probe Request* frames on each one of the available channels. APs respond by sending *Probe Response* frames. Having collected these frames, the STA proceeds with the second phase, the *decision* phase, where it decides which AP it will associate with. According to the standard [1], AP selection is based on the Received Signal Strength Indication (RSSI). A STA simply selects the AP from which it has received the strongest signal during the scanning process. In the third phase, the STA has to follow the *authentication* process if the selected AP follows some authentication mode. Finally, the STA sends an *Association Request* frame to the selected AP and sequentially the AP responds with an *Association Response* frame. If the Association Response frame is received with a "successful" status value, the STA is now associated with the selected AP. Moreover, the standard provides for dynamic re-associations, also termed as handoff procedures, in the case that a mobile node moves away from its associated AP.

Efficient association of a STA with the appropriate AP is a challenging problem of great importance. The overall network performance experienced by end users is greatly affected by association decisions, as all frames exchanged in the resulting BSS, are either transmitted from the selected AP (downlink), or are destined to it (uplink). How-

ever, evaluation of potential associations based on end user performance is a rather complex task, because several factors, such as topology and channel characteristics, as well as interference and traffic conditions, have to be jointly considered. In this work, we propose innovative association and handoff mechanisms, which are able to estimate potential throughput performance among available associations, by taking into account the aforementioned factors in a joint way.

The rest of the paper is organised as follows. In the remaining of this section the state of the art related work is presented and a summarisation of our contribution follows. A detailed analysis of our metric definition follows in section 2. Details about the proposed mechanisms and their implementation are provided in section 3. In section 4, we describe the configuration and present the conducted testbed experiments that evaluate the implemented mechanisms under realistic settings. Finally, in section 5, we present conclusions and discuss our future work.

2.1.1 *Related Work*

The intuition behind the standard technique of selecting APs that offer higher RSSI values is related to the fact that effects like path loss and multi-path fading have a lower impact on performance degradation, as the distance between the communicating nodes is decreased. However, the performance of this rather simple AP selection policy has been extensively studied and it is well known that it leads to inefficient use of the network resources [8],[9]. In addition, due to the asymmetric nature of the wireless medium, this policy becomes unsuitable, as RSSI is an indicator just for the downlink channel and not for the uplink. An association mechanism considering signal to interference and noise ratio (SINR) per connection, as well as asymmetric traffic flows was proposed in our previous work [10]. Although this approach managed to take into account the asymmetric nature of the wireless medium, thus offering significant improvement, it was restricted in evaluating potential associations based only on the assessment of channel conditions. In general, policies that are based solely on consideration of channel conditions may lead to load imbalance among APs, as it is possible for closely spaced STAs to associate with the same AP, in an effort to exploit from the offered channel robustness.

One of the major directions pursued among relevant works has been the proper balancing of load among APs. The authors in [11], proposed an AP selection policy that estimates AP load based on instantaneous measurements of the transmission rate and the fraction of time an AP acquires the channel for transmissions. However, the proposed model faces the disadvantage of considering only downlink traffic and therefore assumes that channel contention exists only among APs. Another common assumption

of works on the field has been to denote AP load as a factor reflecting the AP's inability to satisfy the requirements of its associated users [8],[12]. The approach followed in [13], bases association decisions on a metric denoted as airtime cost, which considers both uplink and downlink traffic, as well as AP load related to the number of associated STAs. The above approaches share the common characteristic of considering only the effect induced by transmissions of associated STAs in the estimation of AP load. However, since the IEEE 802.11 MAC layer is based on channel contention, the efficiency of an AP does not only depend on the impact induced by transmissions of associated STAs, but also on transmissions of other neighboring STAs and APs that operate on the same channel. Trying to address this issue, the authors in [14] consider AP load over all neighboring nodes. The new scheme incorporated the effects imposed by both associated and contending nodes in the performance estimation procedure of the proposed algorithm. However, this approach was restricted in considering only downlink transmissions and setting fixed transmission rates, neglecting the importance of rate adaptation mechanisms.

All the aforementioned approaches follow the assumption of fully saturated traffic, which considers that all network users constantly either transmit or receive frames. In [15], the authors suggest that APs should assign an activity level estimator to their associated STAs, based on observations of their traffic intensity. Nevertheless, this approach does not manage to characterise the traffic activity of neighboring nodes that belong to adjacent cells, although such nodes contend for channel use or even interfere with transmissions of the cell under consideration. Moreover, the proposed scheme requires the operation of an Inter-AP coordinating protocol that provides for collection of activity estimations regarding all network STAs and feeding of such information to a central entity that decides about optimal associations. However, such centralised approaches can only apply to centrally managed deployments, which is not consistent with the distributed structure of the standard association procedure.

One more issue that has been overlooked in the association process, is the effect of interference caused of hidden node terminals, which is inherent in dense WLANs. In a later work in [16], a metric is proposed, that comprises contention and interference as well. The authors, trying to estimate the effect caused of interfering transmissions, define a factor that captures the error probability due to collisions, considering it as a value proportional only to the number of associated and neighboring STAs. The main disadvantage of this approach is that transmissions of APs are not considered as potential interference and thus efficient associations cannot be guaranteed in downlink scenarios.

Another important direction that has also been extensively investigated, primarily aims at distributing throughput among network users in a fair way. In [17], the authors consider the objective of achieving network-wide proportional fairness by using association control. The work in [18] presents a distributed algorithm that performs joint association, rate and contention control, according to the proportional fairness principle. The recent research study, in [19], focuses on balancing the aggregate network performance, while providing fair throughput distribution to network users. The aforementioned approaches identify that fair throughput allocation can dramatically reduce aggregate network throughput and propose corresponding solutions to tackle this issue. However, these schemes do not provide lightweight implementation of the proposed algorithms that can be evaluated under real world network scale and settings and thus are not able to realistically characterise the tradeoff between fair throughput distribution and aggregate throughput maximisation. An innovative scheme that enables fair sharing of channel time by leveraging the idea of network virtualization is presented in [20], but this work focuses on presenting the algorithm implementation details and does not apply the developed mechanism to the problem of association control.

2.1.2 *Our Contribution*

The key novelty of our approach is that it results in comprehensive metrics for user association that capture both contention from "one-hop" and interference from "two-hop" neighbors, not as individual parameters but in a joint manner. The proposed metrics consider the impact of each neighboring node proportionally to its configured PHY-layer transmission rate and its anticipated traffic activity. The resulting algorithms differentiate from relevant approaches in the literature, as their design focuses on the special case of 802.11 infrastructure networks, efficiently discriminating between end user performance on the uplink and downlink channels. We verify the applicability of the proposed mechanisms by integrating them into the 802.11 protocol and evaluate their performance in commercial equipment and under realistic scenarios in a real-life testbed environment.

As a first contribution, we encapsulate in our user association metrics the effect of contention. In contradiction to the aforementioned approaches, we state that AP load should be considered over all neighboring nodes, due to the shared nature of the medium. The IEEE 802.11 MAC protocol provides all compliant nodes with the same chance to access the medium and transmit frames in the long term. As a result, every node in a WLAN shares the medium with all neighboring nodes, not only with APs on downlink or associated STAs on uplink. Moreover, due to the multi-rate capability

at the PHY layer, supported by rate adaptation mechanisms, the transmission duration of a frame depends on the transmission rate selected by the transmitter. The combined effect of the shared medium, in accordance with the multiple PHY rates used, can cause the well known 802.11 "anomaly phenomenon", where low transmission rate STAs negatively affect high bit rate ones [21]. In this work, we take a step further than previous approaches and take into account transmissions of all neighboring nodes (both APs and STAs), in accordance with the assigned PHY transmission rates, in order to realistically estimate contention levels discretely for the downlink and uplink channels, thus extending the definition of AP load.

Another key contribution of the proposed scheme is its ability to adapt to varying traffic conditions. Contrary to relevant approaches, we perform activity estimation at each individual network node, through direct monitoring of the traffic generation rate at the MAC layer transmission queue. Moreover, we propagate this information through neighboring WLANs, by using specially generated for this purpose broadcast frames. This approach enables for online estimation of the prevailing traffic conditions in adjacent cells in a distributed way.

As a third contribution we integrate the effect caused of active hidden terminals. Due to the shared medium, transmissions of interfering hidden nodes can cause collisions and erroneous receptions, that lead to decreased packet delivery ratios (PDR) at the receivers. Counter to relevant approaches, we incorporate in our proposed metric the effects of contending and interfering nodes in a disjoint way. Moreover, we estimate interference levels based on transmissions of all neighboring nodes (both APs and STAs) and arrive at different metrics properly adapted to uplink and downlink channel transmissions.

Our mechanism, integrating all the above features, results in algorithms proposed for both the association and handoff procedures. Through experiments, we present how users that target maximisation of their own throughput, significantly impact aggregate network throughput. Inspired by these observations, we propose two discrete association strategies that target maximisation of individual user throughput and aggregate throughput performance accordingly. We show how the proposed measurement propagation procedure can be readily incorporated in the IEEE 802.11k amendment [22] and moreover move one step further than simulation and implement the proposed algorithms using the Mad-WiFi open source driver [23]. Through extensive testbed experiments, we show how both individual user and aggregate network throughput evolve, as the number of network nodes and injected traffic increase and thus manage to highlight the tradeoff between aggregate throughput maximisation and fair load balancing.

2.2 SYSTEM MODEL AND METRIC DEFINITION

We consider an IEEE 802.11 based WLAN that consists of a large number of APs and STAs. We use $\mathcal{M}$ to denote the set of APs that define a network coverage area. Each $AP_j \in \mathcal{M}$ operates on a single predefined channel, which belongs to the set of non-overlapping channels that the operating band offers.The coverage areas of multiple APs may be overlapping. Within the network coverage area resides a set of mobile STAs, denoted by $\mathcal{N}$, which tend to stay in the same physical locations for long time periods. At any time instant, a $STA_i \in \mathcal{N}$ chooses to associate with a single $AP_j \in M_i$, where M_i denotes the set of APs that operate in the vicinity of STA_i. We use N_j to denote the set of STAs, which are associated with AP_j. Each node of the network $n \in M \cup N$, has a set of neighbors, which reside in its sensing area and operate on the same channel with n . This set of "one-hop" neighbors, that can be either APs or STAs, is denoted by A_n.

Based on the discussion in the preceding section, we notice that the throughput experienced by a node in an IEEE 802.11 network depends apart from channel quality also on the transmission of frames by neighboring nodes. The authors of [24], based on the well known analysis of Bianchi [25], have shown that when there are multiple transmitters using different PHY rates while transmitting, within the contention domain of node n that uses a PHY rate of R_n, then each node of the network enjoys an equally shared value of throughput, which is approximated as:

$$T_{ij} = \frac{1}{\frac{1}{R_{ji}} + \sum_{k=1}^{|A_n|} \frac{1}{R_k}} \tag{1}$$

where R_{ji} and R_k denote the PHY rate used during transmissions performed by AP_j and each node $k \in A_n$ accordingly. This equation ignores the overhead resulting from the 802.11 MAC mechanism. The deficiency mentioned here does not impact the effectiveness of our approach, as we do not aim at calculating the actual end user throughput, but rather focus on deciding about associations that favor throughput performance. Moreover, the above equation considers saturated traffic conditions and requires that all traffic flows consist of equal packet lengths. Throughout this paper, we follow the assumption of equal packet lengths, but later in our analysis we transform the above equation, in order to capture realistic varying traffic conditions. We now modify equation (1), which refers to the general case of networks consisting of multiple flows that are generated between 802.11 compliant devices, in order to conform with the special case of infrastructure 802.11 networks.

2.2.1 *Contention Effect*

We start by considering only saturated downlink traffic, while later in our analysis we consider uplink channel transmissions as well. We assume the usual case, where an $AP_j \in \mathcal{M}$ has $|N_j|$ associated STAs and $|A_j|$ "one-hop" neighboring nodes contending to capture the channel. An AP has an equal probability among its contending nodes to capture the medium for its own transmissions and in each such instant, it uses the medium to transmit to only one of its associated STAs. Under the assumption that the amount of traffic destined to each STA $\in N_j$ is equal and the upcoming association of STA_i with AP_j, we estimate the equally shared value of transmitted bits destined to each STA $\in N_j$ as follows:

$$T_{ij}^{down} = \frac{1}{(\frac{1}{\overline{R_j}} + \sum_{k=1}^{|A_j|} \frac{1}{R_k}) \cdot (|N_j| + 1)} \tag{2}$$

where $\overline{R_j}$ denotes the mean PHY rate used by AP_j and calculated as the average value of the different rates it uses while transmitting to each STA $\in N_j$. According to equation (2) the service rate of AP_j is equally distributed among its $(|N_j + 1|)$ associated STAs.

2.2.2 *Hidden-Node Effect*

As previously stated, these transmissions are still subject to frame errors due to interference at the receivers' side. The main factor that generates interfering conditions in 802.11 networks is the effect of hidden terminals, which appears very often in dense WLANs. In a simple downlink scenario with one AP_j and one associated STA_i, the set of nodes that appear hidden to transmissions of AP_j, consists of nodes existing in the "one-hop" neighborhood of STA_i, which also do not belong in the "one-hop" neighborhood of AP_j. We call this set of nodes as the "two-hop" neighbors of the transmitter, denoted as B_j, equal to the relative complement of set A_j in A_i ($A_i \setminus A_j$). Since the "two-hop" neighbors of the transmitter AP_j are not able to sense its ongoing transmissions, collisions occur, leading to decreased packet delivery ratio (PDR) and consequently result in throughput decrease. The negative effect of the hidden-terminal phenomenon is proportional to the number of active "two-hop" neighbors ($|B_j|$) and their transmissions duration. We capture this effect to model performance decrease by modifying expression (2) as follows:

$$T_{ij}^{down} = \frac{1}{(\frac{1}{\overline{R_j}} + \sum_{k=1}^{|A_j|} \frac{1}{R_k}) \cdot (|N_j| + 1) + \sum_{l=1}^{|B_j|} \frac{1}{R_l}} \tag{3}$$

2.2.3 *Traffic Intensity Estimation*

In most cases, the assumption of saturated traffic is not realistic. Practically, nodes run different applications that generate traffic with varying rates. In order to model realistic scenarios, each transmitter should be properly characterized according to the traffic patterns that it follows.

To this aim, we define for each node $n \in M \cup N$ an activity indication factor, which is denoted as f_n. Each node n measures the rate of packets arriving at its transmission queue during a constant time interval and thus it manages to capture its own packet arrival rate (λ_n). Moreover, node n estimates its own affordable service rate (μ_n) using expression (1), which approximates the maximum rate at which packets may leave its queue for transmission. In the case that the rate of packets arriving at the transmission queue is higher than the rate at which packets leave the queue, only the number of backlogged packets increases, while the traffic injected in the network remains constant. Based on the above, n estimates its maximum affordable traffic rate, by setting its activity indicator f_n, as follows:

$$f_n = \min\{\lambda_n, \mu_n\} \tag{4}$$

Supposing proper periodic messaging of traffic intensity information among neighboring nodes, each node $n \in M \cup N$ is able to detect its active "one-hop" and "two-hop" neighbors and their corresponding activity indicators. In order to acquire long term estimations of traffic intensity per node, the reported value of the activity indicator factor is calculated as the average value during the last periodic reporting interval. Detailed description about the message exchange procedure that is followed by our implementation is provided in the following section. We now use activity indicators to properly assign weights to the impact caused by each node to throughput performance and transform expression (3), as follows:

$$T_{ij}^{down} = \frac{1}{(\frac{f_j}{\overline{R_j}} + \sum_{k=1}^{|A_j|} \frac{f_k}{R_k}) \cdot (|N_j| + 1) + \sum_{l=1}^{|B_j|} \frac{f_l}{R_l}} \tag{5}$$

where $\overline{R_j}$ is now calculated considering the amount of traffic destined to each individual STA that is associated with AP_j.

A similar approach can be followed for uplink communication as well, where frames are transmitted by each STA and destined to the specific AP it is associated with. The transmitter, STA_i in this case, shares the medium with its "one-hop" neighbors (A_i),

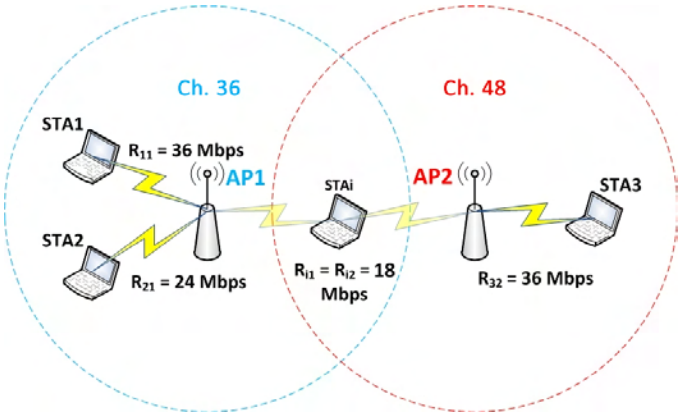


Figure 4: Topology of initial testbed experiments

while its "two-hop" neighbors (B_i) are the nodes that are located in the AP's neighborhood but not in the neighborhood of the STA ($A_j \setminus A_i$). For the uplink case, we arrive at the following expression:

$$T_{ij}^{up} = \frac{1}{\frac{f_i}{R_{ij}} + \sum_{k=1}^{|A_i|} \frac{f_k}{R_k} + \sum_{l=1}^{|B_i|} \frac{f_l}{R_l}} \tag{6}$$

Considering expressions (5) and (6), STA_i under association, is able to estimate the throughput that the upcoming association with each $AP_j \in M_i$ may offer, both on uplink and downlink. Consequently, STA_i decides to associate with the $AP_j \in M_i$ that offers the highest calculated metric values in an effort to maximize its own throughput (*Selfish strategy*).

In order to investigate the estimation accuracy of the proposed metrics, we conduct realistic testbed experiments and compare the estimated throughput results with the experimentally obtained ones. Moreover, through these initial experiments, we also aim at presenting how sequential associations affect performance of preexisting active flows and aggregate network performance. In the next section, we refrain from presenting extensive results regarding our initial experimentation efforts, but present a detailed analysis of a representative scenario, which highlights the main conclusions that were reached and motivated our design choices.

2.2.4 *Initial Experiments on Realistic networks*

The experimental setup consists of 2 APs that operate on the orthogonal channels 36 and 48 of the 5 GHz band that experience zero external interference in our testbed premises. Fig. 4 represents this experimental topology, where STA1 and STA2 are associated with AP1, using PHY rates denoted by R_{11} (36 Mbps) and R_{21} (24 Mbps)

Throughput/Node	Throughput if STA_i associates with	
Node	AP1 (Mbps)	AP2 (Mbps)
STA_i	6.4	8.4
STA1	7.4	12.3
STA2	5.6	10.1
STA3	29.4	10.3
AGGREGATE	48.8	41.1

Table 1: Realistic throughput performance based on testbed experiments.

accordingly, while STA3 is associated with AP2 and transmits using PHY rate R_{32} (36 Mbps). The station under association, STA_i is statically assigned the fixed PHY rate of 18 Mbps in both cases ($R_{i1} = R_{i2} = 18$ Mbps).

We measure the experimental throughput performance on uplink, considering saturated flows generated by each STA and list the collected results in Table 1, where each column represents the results obtained in the case that STA_i decides to associate with AP1 or AP2. Based on the conducted testbed experiments, we observe that association of STA_i with AP1 and AP2 leads STA_i to experience throughput performance of 6.43 Mbps and 8.4 Mbps accordingly. We also estimate performance of both association decisions, based on the evaluation of expression (6), and present the obtained results in Table 2. Based on the proposed metrics, we estimate that the upcoming association of STA_i with AP1 will result in performance of 8 Mbps, while association with AP2 will result in throughput of 12 Mbps. Following the proposed approach, STA_i would associate with AP2 in order to maximize its own throughput (*Selfish* strategy).

Considering the experimental performance (Table 1) of the rest testbed nodes, namely STA1, STA2, STA3, we obtain the results of 7.4 Mbps, 5.6 Mbps and 29.4 Mbps, if STA_i associates with AP1, while in the case that STA_i associates with AP2 the results are 12.3 Mbps, 10.1 Mbps and 10.3 Mbps accordingly. When considering the aggregate network throughput performance, we notice that the Selfish strategy, which leads STA_i to associate with AP2, results in performance of 41.1 Mbps, which is lower than the performance of 48.8 Mbps that is offered by the association with AP1.

In addition, we observe that the experimentally verified aggregate performance degradation is also detected through the aggregation of the results that estimate performance of each individual node in Table 2. More specifically, we estimate that the aggregate network throughput performance will reach the value of 60 Mbps, in the case that STA_i associates with AP1, while association with AP2 will result in throughput of 52.8 Mbps. Direct comparison between the realistic and the estimated results that were collected in this experiment, clearly shows that the estimated performance approximates well the

Throughput/Node	Throughput if STA_i associates with	
Node	AP1 (Mbps)	AP2 (Mbps)
STA_i	8	12
STA1	8	14.4
STA2	8	14.4
STA3	36	12
AGGREGATE	60	52.8

Table 2: Estimated throughput performance based on evaluation of the proposed metrics.

realistic testbed performance. The performance difference is mainly due to the fact that the proposed metrics ignore the overhead induced by the IEEE802.11 MAC mechanism. Moreover, we observed that certain topological settings were also able to significantly impact the resulting performance. Having conducted extensive testbed experiments that considered various topologies, we experimentally verified that the accuracy provided by the proposed metrics is capable to drive association decisions that target maximisation of individual node throughput performance.

As a second outcome of our initial testbed experiments, we observed that nodes featuring lower channel contention (STA3 in Fig. 4) and as a result experience higher performance, will suffer significant throughput degradation, in comparison with nodes that feature higher contention (STA1 and STA2 in Fig. 4). Therefore we arrived at the conclusion that it is important to develop appropriate association mechanisms, where each node will be taking into account not only its individual performance, but the overall performance of the network as well. From now on, we will refer to the approach that considers throughput maximisation of the node under association as the *Selfish* strategy and thus distinguish it from the approach that targets maximisation of aggregate network performance, which we will refer to as the *Altruistic* strategy.

2.2.5 *Altruistic User Strategy*

As demonstrated through the previous experiment, a suitable user association policy has to consider social objectives rather than applying Selfish strategies myopically, in order to be able to define associations that maximize aggregate network performance. Given the fact that association decisions are taken by STAs in a distributed way, we envision appropriate metrics that estimate aggregate network performance locally in the neighborhood of the STA under association. In this perspective, we propose that each STA takes association decisions based on local information, regarding its "one-hop" neighbors and locally maximizes the estimated aggregate performance of all neighboring nodes. Based on the above, STA_i under association, estimates aggregate network performance for each $AP_j \in M_i$, as follows:

$$T_{ij}^{al} = T_{ij}^{sel} + \sum_{k=1}^{|A_j|} T_k^{sel}, \tag{7}$$

considering both its Selfish performance, as well as the performance of all neighboring nodes, where T_k^{sel} denotes the estimation of throughput following the Selfish strategy, through expressions (5) and (6), for the downlink and uplink cases accordingly.

2.3 PROPOSED ALGORITHMS

The proposed mechanism requires each network node $n \in M \cup N$ to propagate information describing its PHY rate and activity indicators, as well as its "one-hop" neighbors. This information exchange procedure has to be periodical, so that all nodes get up to date information about the prevailing channel and traffic conditions. In the rest of this work, we use t_r to denote the period of this information exchange procedure. Moreover, the proposed mechanism requires the APs to propagate information describing the number of already associated users ($|N_j|$) and the mean PHY rate ($\overline{R_j}$) used for downlink communications, as calculated upon observation, according to the percent of traffic that is destined to each associated STA. During the scanning period, STA_i scans all the available channels to collect the information required for estimating performance considering every $AP_j \in M_i$.

The proposed measurement propagation procedure can be readily incorporated in the IEEE 802.11k amendment, which provides frameworks for collecting and disseminating radio resource measurements. The objective of 802.11k is to optimise the operation of standard 802.11 networks, by enabling STAs to optimally assess the radio environment in which they operate. To fulfil its goal, the current 802.11k draft defines different types of measurements, which enable compliant network nodes to collect information about in range APs (*Beacon* report frame), and about neighboring STAs (*Frame* report, *Hidden Node* report and *STA Statistics* report frames). The *STA Statistics* report frame, which is transmitted by STAs, provides measurements that include the average access delay and number of associated STAs per AP. In the same perspective, we introduce the transmission of a special broadcast frame, which contains all the information required in order to assess the activity of its "one-hop" neighbors. The introduced measurement frame, which we denote as *Neighbor Report* frame, enables STAs under association to estimate potential throughput performance, by not only exploiting from measurements collected by neighboring STAs, but also by adjusting the collected information based on the effect that its own "one-hop" and "two-hop" neighbors induce.

2.3.1 *Association Mechanism*

The analysis presented in the preceding section, concludes in expressions that estimate throughput performance of the *Selfish*, as well as the *Altruistic* approach both on uplink and downlink. In the case that the considered strategy is the *Selfish* one, the application of the *Selfish* metric in the AP decision procedure is straightforward. In essence, STA_i decides to associate with the $AP_j \in M_i$ that provides the maximum calculated *Selfish* metric. On the other hand, direct application of the *Altruistic* strategy faces two inherent problems that were identified during experimentation. The first problem, which results in unequal bandwidth sharing, is identified in cases of BSSs that consist of just one STA that uses one of the highest available PHY rates (48 or 54 Mbps). In such cases, all neighboring STAs that follow the *Altruistic* approach, will try to avoid associating with the specified APs, in order not to decrease the high aggregate network performance that is yielded due to transmissions of the STAs that use the high PHY rates. The second problem is identified in scenarios where the number of network STAs increases dramatically. In such scenarios, as the number of STAs increases above a certain limit, potential associations tend to be assigned metric values that fluctuate around constant values, thus limiting the range and variety of available associations. Having conducted extensive testbed experiments to evaluate the performance of the proposed approaches, we managed to identify the performance limits of both approaches, considering both throughput maximisation and fair load balancing, as our performance metrics. Based on the obtained results, we observed that direct application of the *Altruistic* strategy results in significantly unfair throughput distribution and thus we decided to design a lightweight hybrid approach that is built on the *Altruistic* strategy and is also able to tackle with the two aforementioned issues. More specifically, we introduce a throughput threshold (T_{TH}) that triggers the use of the Selfish metric, instead of the *Altruistic* metric, in cases that the estimated performance difference for two discrete APs is lower that T_{TH}. Application of this plain heuristic, enables the *Altruistic* approach to overcome both of the deficiencies that were identified in the scenarios mentioned above. A brief description of the implemented Association algorithm follows:

- **STEP 1:** Transmission of special *Neighbor-Report* frames by each network node every t_r seconds. Transmission of *Beacon* frames every *Beacon* interval and *Probe-Response* frames on demand.
- **STEP 2:** Periodic repetition of Scanning procedures by STA_i on each available channel, to characterise A_i and B_i neighbourhoods .
- **STEP 3a Selfish:** Calculation of metric value T_{ij}^{sel}, for each potential $AP_j \in M_i$, using expressions (5) and (6) for uplink and downlink accordingly.

- **STEP 3b ALTRUISTIC:** Calculation of metric value T_{ij}^{al}, for each potential $AP_j \in M_i$, using expression (7) for uplink and downlink accordingly. If the difference in estimated throughput performance between two potential associations is lower than T_{TH}, the metric T_{ij}^{sel} is used, as calculated using expressions (5) and (6), for each case.
- **STEP 4 :** STA_i associates with the AP_j that offers the highest calculated metrics for each strategy.

2.3.2 *Handoff Mechanism*

In addition, we incorporate the proposed metrics in the handoff procedure as well. A handoff, in the IEEE 802.11 standard, is the process that enables a STA to change the AP it is associated with, in the cases that it detects degradation of RSSI. As previously explained, several more factors, than the signal strength, affect quality of communication, which fact necessitates the application of sophisticated metrics that jointly consider the aforementioned factors, in order to result in efficient handoff decisions.

Considering the metrics proposed in the previous section, STA_i is constantly able to monitor the throughput performance that its current association offers and consequently decide about whether a handoff to another available AP is required. STAs need to perform *scanning* operations on the rest available channels, in order to estimate performance of potential associations with APs that operate on other channels. We use the T_{TH} threshold to trigger such scanning procedures, in cases that the difference between the performance yielded upon the STA's initial association decision and the performance currently experienced by the same association, exceeds this threshold. A brief description of the implemented handoff algorithm follows:

- **STEP 1a Selfish:** Constant monitoring of performance achieved by current association, through the calculation of metric value T_{ij}^{sel}, using expressions (5),(6), for uplink and downlink accordingly.
- **STEP 1b ALTRUISTIC:** Constant monitoring of performance achieved by current association, through the calculation of metric value T_{ij}^{al}, using expression (7), for uplink and downlink accordingly.
- **STEP 2:** If the difference between current and initial performance exceeds T_{TH}, scanning procedures are executed.
- **STEP 3 :** STA_i performs handoff to the AP_j that offers the highest calculated metrics.

2.3.3 *Implementation Details*

As the features provided by the the IEEE 802.11k amendment for collecting and disseminating radio resource measurements, are currently not implemented in open-source drivers, we decided to implement the required measurement exchange mechanisms from scratch, in the MAD-WiFi open source driver. In this section we describe the key challenges encountered during the implementation of the proposed algorithms. One of the key features that the proposed mechanisms require is the ability of all nodes to receive information about ongoing transmissions from neighboring BSSs that operate on the same channel. However, all frames 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. As a result, only unicast and multicast frames that are destined to the adapter's MAC address along with broadcast frames can be captured. In order to accomplish communication of information among neighboring BSSs, we generated a custom 802.11 broadcast frame, which includes the information that the *Neighbor Report* frame introduces, and more specifically the PHY rate used in the latest transmission and the activity indicator, as computed using expression (4). The reported activity indicators are properly calculated as the average value over the duration of the last *Neighbor Report* interval (t_r), in order to avoid temporal fluctuations caused due to transmission of bursty traffic.

The introduced *Neighbor Report* frame, which is periodically transmitted by each network node n, also carries information describing its "one-hop" neighbors ($k \in A_n$), including their PHY rates (R_k) and activity indicators (f_k) accordingly, based on the reception of their corresponding *Neighbor Report* frames. The "two-hop" neighbors of each node n ($k \in B_n$), are calculated as the relative complement of each set A_k in A_n ($A_n \setminus A_k$) accordingly. Based on the above, node n gets updated about the prevailing contention and interference conditions of each channel, every t_r seconds.

A third modification of the driver included the extension of the *Beacon* and *Probe-response* frames transmitted by the APs, by adding two extra fields that contain the calculated $|N_j|$ and $\overline{R_j}$ values. These frame extensions do not affect the normal operation of the 802.11 protocol, as according to the standard they both feature a dynamic part that supports extension. Further, in order to estimate the PHY rates (R_{ji} and R_{ij}) that STA_i will use in the case of association with AP_j, we implemented a PHY rate estimation procedure that considers the received signal strength of Beacon and Probe-Response Frames that are transmitted by each $AP_j \in M_i$. Our final modification was performed in the scanning procedure, where we set the interval that STAs have to remain on each channel equal to the *Neighbor Reports'* interval (t_r).

2.4 EXPERIMENTAL EVALUATION

In order to evaluate the performance and study the behaviour of the implemented association and handoff mechanisms under real conditions, we used a large scale programmable testbed of wireless nodes, called NITOS.

2.4.1 *NITOS Testbed*

NITOS (Network Implementation Testbed for using Open Source platforms) is a large scale wireless testbed that has been designed to support evaluation of protocols and applications in real world settings. NITOS is *remotely* accessible and currently consists of 50 wireless nodes, outdoor located in a non-RF-isolated environment. The nodes are equipped with 2 wireless interfaces using the AR5414 - Atheros 802.11a/b/g compatible chipset, which runs the MAD-WiFi open source driver. Users can perform their experiments by reserving slices (nodes, frequency spectrum) of the testbed through NITOS scheduler, that together with the OMF (cOntrol and Management Framework) experiment orchestration framework [26], support ease of experimentation.

2.4.2 *Measurement Methodology*

As NITOS is outdoor located in a non-RF-isolated environment, in order to provide for a proper experimentation setup, we conduct all our experiments in channels of the 5 GHz band, which experiences zero external interference in the testbed premises. The throughput performance of the experiments, is measured by using Iperf, which is a powerful tool for traffic generation and measurement. 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. In our experiments, we use a custom version of the Iperf application, which is orchestrated in an automatic way with OMF and enables the collection of measurements through the OML (OMF Measurements Library) Library. In each experiment, we first activate the APs and subsequently the STAs one by one, introducing a fixed time interval of 5 seconds. When the first phase of the initial associations ends, all STAs that detect degradation of the initially calculated performance, start checking for potential handoffs, by enabling the handoff mechanism. To remove any random effect and short-term fluctuation, we run each experiment 5 times and each run lasts for 10 minutes. The final results, presented in the next section, show the average values obtained during the execution of multiple runs of the same experiment. Finally, in order to evaluate the fairness in throughput distribution among network nodes, we also calculate the Jain's fairness index [27] for each conducted experiment. We use the Jain's fairness index to

rate the fairness of throughput distribution between n users, where the result ranges from $\frac{1}{n}$ (worst case) to 1 (best case), when all users receive the same allocation.

2.4.3 *Experiments*

Based upon the testbed described above, we conduct numerous experiments and below we present and analyse the obtained results. The first series of experiments has been designed to evaluate performance of the *Selfish* strategy, where each user myopically selects the AP that maximizes its own throughput. We perform these experiments in two discrete phases, where in the first one we use the unmodified MAD-WiFi driver that follows the RSSI approach, while in the second phase we use the modified version of the driver that implements the *Selfish* approach. In order to examine the performance of the *Altruistic* approach, where the objective followed is the maximisation of aggregate network throughput, we setup the second series of experiments. These experiments are executed in three phases and present a joint performance comparison between the RSSI, the *Selfish* and the *Altruistic* approaches, where in each experiment execution phase we use the specific driver version that implements the corresponding approach. We run the aforementioned experiments either on downlink or uplink, but we do not run experiments measuring combined throughput on purpose, as we expect the performance to be relevant. The third series of experiments that consider the overhead induced by the proposed mechanisms and provide feedback about the appropriate values regarding the settings of our algorithms, were conducted prior to the first two series of experiments. Based on the results obtained through this last series of experiments, we set the *Neighbor Report* interval equal to 1 sec. and the Handoff threshold H1 equal to 10%, for all the scenarios considered in the first two series of conducted experiments. Moreover, in order to demonstrate the ability of the proposed algorithms to dynamically adapt to the prevailing channel conditions, we decided to enable the driver Rate Adaptation algorithm, instead of conducting fixed-rate experiments.

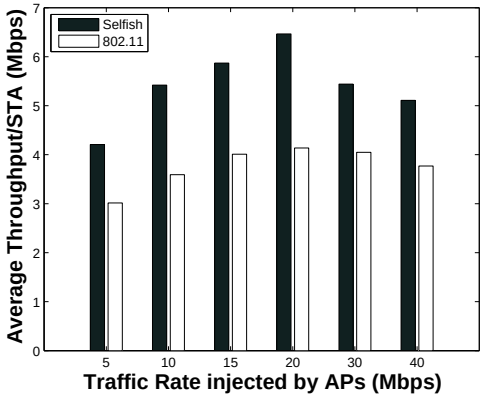
2.4.3.1 *Selfish Strategy experimentation on Downlink*

In this first series of experiments, we set up a network that consists of 3 APs and 14 STAs, where AP1 operates on Ch. 36, while both AP2 and AP3 operate on Ch. 48 of the 5GHz band. Figures 5a and 5b provide a representation of the testbed's topology used to implement this experimental scenario, while also demonstrating the association decisions of STA based on the RSSI and *Selfish* approach accordingly. The actual testbed topology consists of 3 testbed nodes that run in AP mode and 7 testbed nodes that are configured to operate in STA mode, with both of their wireless cards, resulting in a network of 14 STAs. We perform these experiments in two discrete phases, where in the first one we use the unmodified MAD-WiFi driver that follows the RSSI approach,

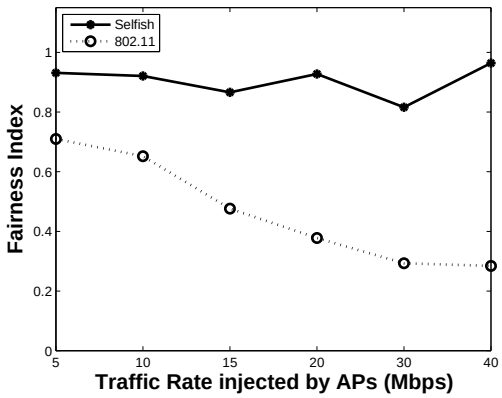
(a) Testbed Topology and resulting associations following the RSSI-based approach

(b) Testbed Topology and resulting associations following the "Selfish" approach

Figure 5: Selfish Strategy experimentation on Downlink



(a) Average Throughput Performance



(b) Jain's Fairness index

Figure 6: Selfish Strategy experimentation on Downlink

while in the second phase we use the modified version of the driver that implements the *Selfish* approach. In both figures, the relative distance between STAs and APs represents the actual testbed topology while a solid line is used to illustrate the association of the first interface of each node that operates in STA mode, while a distinctive line is used to represent the association of the second interface. The APs transmit on downlink, generating UDP traffic of varying rate through 14 Iperf clients that run simultaneously, while the corresponding Iperf servers run at each STA.

Following the RSSI approach, only 2 STAs associate with AP1 that operates on Ch. 36, while the rest 12 STAs, which are located closer to AP2 and AP3, decide to associate with them and thus operate on Ch. 48. On the other hand, the combined application of the proposed *Selfish* association and handoff mechanisms enables STAs to detect the contention for channel use between AP2 and AP3 and thus leads more STAs to associate with AP1. Fig. 6a presents how the average throughput achieved per STA changes with respect to the load applied, in both cases. The *Selfish* strategy enables APs to deliver substantial higher throughput, in comparison with the standard IEEE 802.11 approach, for all the various considered levels of traffic load. More specifically, the RSSI approach reaches the maximum throughput value when the load reaches the

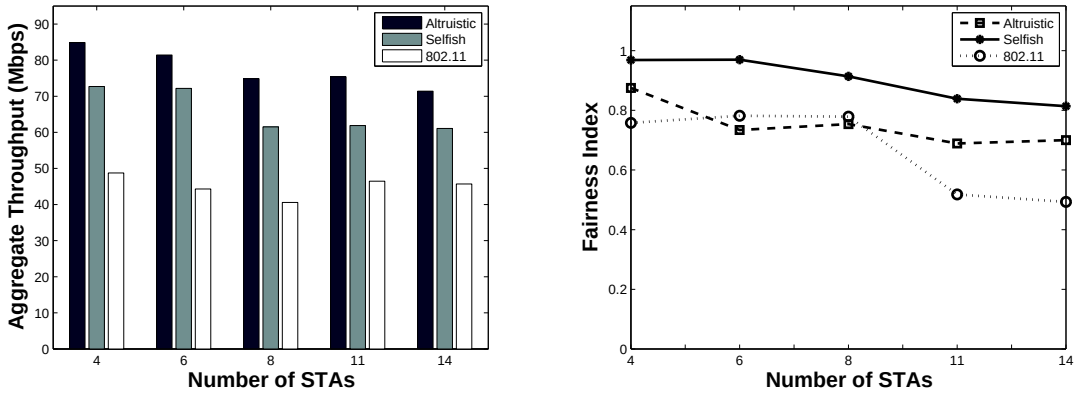
value of 15 Mbps/flow, while our approach reaches the maximum throughput for load value of 20 Mbps/flow, leading to performance increase of 62.5%. Once the traffic rate per flow increases above the value of 30 Mbps/flow, all stations invariably start to witness significant packet drop and throughput deterioration, for both of the approaches followed. However, our scheme still continues to achieve higher performance, in comparison with the RSSI approach.

Another issue of significant importance is the fairness feature that our scheme provides. Fig. 6b presents the Jain's fairness index values calculated for the various cases of applied traffic load, in both of the considered approaches. As portrayed in Fig. 6b, the proposed approach manages to provide nearly fair sharing of available throughput among network users, for all the cases studied. On the other hand, the standard approach, leads to associations that favour only a subset of STAs with high throughput and more specifically STA1 and STA2 that experience low channel contention on Ch. 36, while letting the rest STAs to experience significantly lower performance, as they all operate on Ch. 48. In essence, we conclude that the *Selfish* strategy manages to balance the traffic load of the network not only among the available APs, but also among the available channels of operation, which results in nearly equal sharing of the available throughput among the corresponding receivers.

2.4.3.2 *Altruistic - Selfish Strategies experimentation on Uplink*

The second series of experiments is designed to evaluate the performance of the *Altruistic* approach, in comparison with both the *Selfish* and the RSSI-based ones. For this purpose, we design a network setup that consists of 3 different BSSs, operating on separate channels of the 5GHz band. More specifically, we use 3 APs, where AP1 is assigned Ch. 36, AP2 Ch.40 and AP3 Ch.48 and a varying number of STAs. We consider only uplink scenarios, where the STAs generate saturated UDP traffic destined to the corresponding APs.

In these experiments, we vary the number of STAs that consist the network setup, in order to investigate how each discrete approach scales with networks of different size. The aggregate throughput performance, measured by the APs with respect to the number of STAs, is illustrated in Fig. 7a. It is clearly depicted that both the *Selfish* and *Altruistic* approaches outperform the RSSI approach, in terms of throughput performance, for all the different network sizes studied. More specifically, we observed that for networks consisting of up to 6 STAs, the triggering mechanism of the *Altruistic* algorithm is not yet activated and as a result aggregate throughput maximisation is sought. As the number of network STAs increases above 8, the triggering mechanism is enabled and leads upcoming associations to favour individual node performance, while limit-



(a) Aggregate Throughput Performance

(b) Jain's Fairness index

Figure 7: Altruistic - Selfish Strategies experimentation on Uplink

ing aggregate network throughput. However, the throughput difference that results due to the initial associations is still maintained, even in larger network topologies.

A particular observation is related to the throughput increase monitored during the application of the RSSI approach in the cases of network setups that consist of 11 and 14 STAs. Through careful observation of the experimental logs, we notice that none of the 6 initially enabled STAs, decides to associate with AP3, as this AP is spaced a few meters away and thus offers lower RSSI values. However, while considering larger network topologies that consist of 11 and 14 STAs, we notice that the STAs activated last during the experiment execution, decide to associate with AP3, as it provides higher RSSI values. These observations clearly justify the non-uniform monitored performance increase and moreover demonstrate the heavy dependence of the RSSI approach on the underlying network topology. Contrary to the standard approach, both of the proposed approaches are not only agnostic to topological settings, but moreover are channel contention aware and thus are able to exploit the channel diversity offered by the APs that operate on different channels.

The fairness performance results lead to enlightening conclusions, regarding the distribution of throughput across networks of varying size. As illustrated in Fig. 7b, the *Selfish* approach manages to provide nearly equal sharing of available throughput among network users, for the various considered numbers of activated STAs. In addition, we notice that the RSSI and *Altruistic* approaches present similar Jain's fairness index values for networks that contain up to 8 STAs. The relatively low performance that the *Altruistic* strategy achieves, is mainly due to the fact that it leads only 2 STAs with high PHY-layer rate values (of 48 Mbps) to associate with the same AP and as a result experience high throughput values, due to the prevailing low channel contention

levels. This comes in contrast with the rest low PHY-layer rate STAs, which are distributed among the other 2 APs. On the other hand, the RSSI approach experiences low fairness performance, due to the specific topological settings, which result in associations that favour only a small subset of nodes. However, upon the activation of the rest STAs in the 11 and 14 STAs networks, the superiority of the hybrid *Altruistic* approach against the standard compliant one is distinguished.

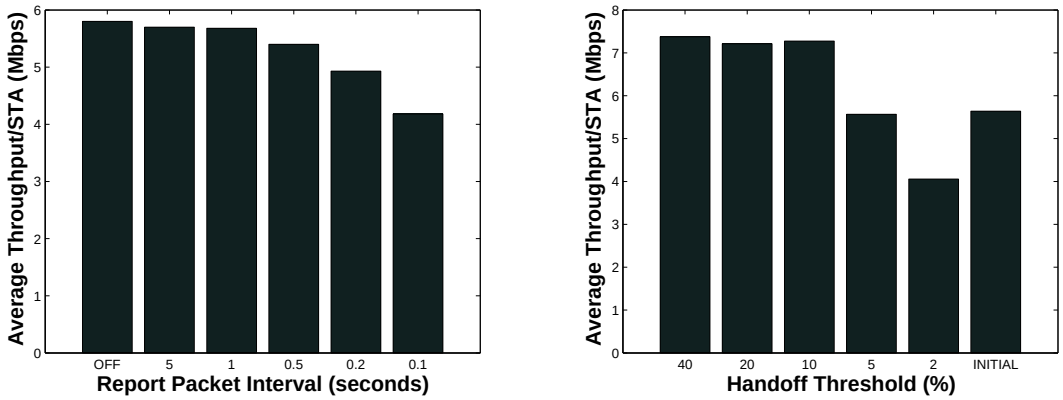
Based on the obtained results, we managed to highlight the tradeoff between network throughput maximisation and fair load distribution and moreover verified that the hybrid *Altruistic* algorithm is able to achieve this tradeoff through proper application of each strategy, under different conditions. Concluding, we remark that both of the implemented algorithms manage to achieve their individual design goals, while also exhibiting better performance in terms of achieved throughput and fairness, compared with the approach specified by the IEEE 802.11 standard.

2.4.3.3 *Overhead Consideration*

In this last series of experiments, we discuss the overhead that the proposed scheme imports. It is known that the period at which the *Neighbor Report* frames are broadcast, exerts crucial influence on system performance. In order to investigate this impact, we vary the transmission period of *Neighbor Report* frames, under a specific network topology, in the experiment that follows first. Another factor that can severely affect performance is the frequency at which handoff procedures are performed. The experiment presented last in this section has been designed to evaluate the performance of the handoff mechanism, under various settings of the T_{TH} threshold.

NEIGHBOR REPORT FRAME INTERVAL EXPERIMENTATION

For the purposes of this experiment, we design an experimental setup that consists of 2 APs and 8 specific STAs that provide links of good quality with the corresponding APs. We reached this decision, in order to provide for a proper measurement setup, so that performance fluctuation would be minimized and even small variations in measured throughput could be detected. We start our experimentation by using the implemented *Selfish* approach and set the frequency of *Neighbor Report* frames transmission equal to the frequency at which *Beacon* frames are transmitted (100ms). As soon as the network has converged to stable associations, we proceed with the next phases of the experiment, where we statically force the initially selected associations and under these settings we vary the *Neighbor Report* frame interval values. In order to compare the proposed scheme with the 802.11 standard approach, we also measure performance in the case that no *Neighbor Report* frames are transmitted.



(a) Neighbor Report Frame Interval

(b) Handoff Throughput Threshold

Figure 8: Overhead related Experimentation

Fig. 8a illustrates how various *Neighbor Report* frame interval (t_r) values affect the resulting throughput performance. We notice that there is no significant decrease in the network performance between the cases that no broadcast frames are transmitted (802.11 standard) and when the interval is set to 1 sec. When the interval value falls between the range of 0.5 to 0.1 secs., the decrease in performance is considerable. Low interval values let the STAs be constantly updated of the current contention, interference and traffic conditions on the operating channel, while on the other hand, such low values induce high overhead. As interval values increase, the resulting overhead can be reduced, but the information about varying conditions may become stale. However, a general optimal operating region of the *Neighbor Report* interval value is far more complicated to predict, as the suitability of this interval highly depends on the variance of contention, interference and traffic conditions.

HANDOFF THRESHOLD EXPERIMENTATION

The aim of this last experiment is to estimate how different values of the T_{TH} threshold impact network performance. For the purposes of this experiment, we setup a new scenario that contains 3 APs and 8 STAs, where AP1 and AP2 operate on Ch. 48, while AP3 operates on Ch. 36 of the 5GHz band. During the initial phase of the experiment, where we use the RSSI based approach, 6 of the STAs associate either with AP1 or AP2 that operate on the same channel. Due to the inability of the RSSI approach to detect channel contention between AP1 and AP2, there exists high potential for throughput improvement through handoffs.

In the second phase, all STAs decide about potential handoffs, based on comparison between the throughput performance of their initial association and the performance that is currently experienced, considering the *Selfish* strategy. At this point, the option

about execution of scanning operations that may result to handoffs is controlled by threshold T_{TH}. A high value of this threshold leads STAs to repeat scanning operations more frequently, which may result in inefficient scanning that generates high overhead, as it causes intervals in normal reception and transmission procedures. Threshold T_{TH} is varied in this experiment, and the resultant throughput is measured and plotted in Fig. 8b. We notice that the first threshold value of 2% causes rather frequent repetitions of scanning procedures and thus results in much lower throughput. When the interval increases to 5% and subsequently to 10%, a significant increase in performance is noticed, due to the decrease in the frequency of inefficient scanning procedures and the resulting performance improvement of the corresponding handoffs. Finally, we notice that the two last values of 20% and 40%, lead to the initially decided associations, as there exist only a few available transitions that can be triggered, under these high threshold values. Having conducted several experiments on various topologies and under varying threshold values, we arrive at the conclusion that optimal values about T_{TH} threshold cannot be predicted in general. This comes from the fact that resulting performance depends heavily on the variation of the several factors that are taken into consideration by the proposed metrics, apart from specific threshold values.

2.5 CONCLUSIONS AND FUTURE WORK

In this work, we have proposed novel association and handoff mechanisms, which are based on innovative metrics that are able to capture the effects of contention and interference in the neighborhood both on uplink and downlink. A key feature of the proposed scheme is that it manages to adapt to realistic traffic conditions. The proposed approach results in two discrete strategies that target the maximisation of individual node and aggregate network throughput performance accordingly. Compared with the standard RSSI-based approach, both of the implemented algorithms exhibit far better performance in terms of throughput and yielded fairness, as evaluated under realistic testbed experiments. As a first extension, we plan on studying the problem of user association and frequency selection [28] jointly, as proposed in [29]. Furthermore, a future objective is to test new mechanisms that jointly perform transmit power and association control, in order to see how the difference in transmission range can affect the "one-hop" and "two-hop" neighborhoods, which in turn affect the corresponding associations.

3

EXPERIMENTATION ON USER ASSOCIATION IN FEDERATED TESTBED ENVIRONMENTS

3.1 INTRODUCTION

Wireless Mesh Networks (WMNs) are currently considered as the default solution for delivering high-speed Internet access to users within the last few network miles in non-urban areas. As a result, the interest of the research community in proposing WMN-related approaches has dramatically increased during the last few years. The inherent inability of simulation models to accurately estimate performance of wireless networks [30], in accordance with the unique characteristics introduced by the complex nature of WMNs [31] have directed research efforts towards implementation approaches and evaluation through experimentation in real world network scale and settings.

However, development of large scale WMN testbeds is a rather challenging task that requires careful design and induces high deployment and maintenance costs. Moreover, as WMNs are usually considered as a promising technology for Internet access provision, experimentation across global scale networks that feature real Internet characteristics is required, in order to acquire experiment results under realistic congestion conditions. Such requirements have led the research community to the decision to develop a global large scale infrastructure that results from the federation of heterogeneous types of networks, such as wired (local, wide-area or optical) and wireless (local, mesh or sensor) networks.

Federation between inherently heterogeneous testbeds introduces several issues that arise due to the difference in the nature of experimental resources, but more importantly due to use of different software frameworks for resource management and control. In this work, we describe the establishment of the federation between two well-known heterogeneous network testbeds, namely the NITOS wireless testbed and the wired PlanetLab Europe (PLE) testbed. The utilization of a common experiment control framework, OMF [26] (cOntrol and Management Framework), and the adoption of the slice abstraction, as the building block for the federation, have been the main keys which made this federation possible.

In order to demonstrate the usefulness of the resulting integrated architecture, we develop and implement an association scheme for WMNs that is aware of end-to-end delay, part of which is generated in the wired section (PLE) and part in the wireless section (NITOS). The implemented mechanism is based on novel association metrics [32] that consider wireless channel contention, which are further enhanced, by taking into account delays incurred in the wired section of the end-to-end path. The evaluation of the proposed mechanism is performed through extensive experiments conducted on the combined network architecture, which results from the federation of the two heterogeneous experimental facilities. In an effort to explore the key differences and potential performance dissimilarities between realistic testbed and simulated environments in performing Mesh Network experimentation, we reproduced several of the testbed experiments in a well-established network simulator.

This paper is organized as follows. In Section 2 we discuss research work related with both association in WMNs and federation of heterogeneous experimental facilities. In Section 3 we describe the architecture of the two heterogeneous testbeds and moreover provide details about the approach followed and the tools used for the establishment of the testbed integration. In Section 4 we analyze and discuss the proposed association approach. The measurement methodology followed in our experiments, as well as a brief experiment description that is OMF compatible are presented in Section 5. In Section 6 we present and comment on the results obtained from the experimental evaluation of the implemented mechanism. Results obtained through the simulation of the real experiments, along with achieved user experience are discussed in Section 7. Finally, in Section 8 we summarize our work, by pointing out conclusions and directions for future work.

3.2 RELATED WORK

3.2.1 *Association in Wireless Mesh Networks*

WMNs are composed of Mesh Routers (MRs), which form the wireless backhaul access network and Mesh Clients (MCs). MRs forward packets acting as intermediate relay nodes and may also provide wireless access services to MCs, in which case they are referred to as Mesh Access Points (MAPs). WMNs also consist of Internet Gateway nodes (IGWs) that provide Internet access to the network, through direct connection to wired infrastructure. MCs associate with a certain MAP in order to access the network and do not participate in packet forwarding.

The affordable cost and ease of deployment of IEEE 802.11 compliant equipment has led the majority of WMNs to be based on conventional IEEE 802.11 devices, although this does not limit the potential application of other standards. According to the IEEE 802.11 standard, which was originally proposed for infrastructure Wireless Local Area Networks (WLANs), MCs perform scanning to detect nearby MAPs and simply select to associate with the MAP that provides the highest Received Signal Strength Indication (RSSI) value. The performance of the standard association policy has been extensively studied [33] in the context of IEEE 802.11 WLANs and it is well known that it leads to inefficient use of the network resources. In WMNs, the entire path between the MC and the IGW is composed of two discrete wireless parts: the single-hop access link between the MC and the MAP it is associated with and the multi-hop backhaul part that connects the MAP with the IGW. As the standard policy considers only factors affecting performance on the wireless access link, its direct application on WMNs becomes inappropriate. As a result, more sophisticated association schemes are required to capture performance achieved in both the access and the backhaul network parts.

Trying to address the issues generated by the unique two-tier architecture introduced by WMNs, several approaches on MAP selection have been proposed in the recent literature. An innovative cross-layer association mechanism that considers not only the access link but also routing in the multi-hop backhaul part is proposed in [13]. The authors in [34] consider also the interaction of physical (PHY) layer transmission rate with the packet size and hop count and propose a signaling mechanism through which information about congestion on both parts is passed from the MAPs to the MCs. In [35], a new metric is proposed that takes into account the impact of 802.11 MAC layer contention on bandwidth sharing and results in accurate link throughput estimations. Another approach, proposed in [36], considers also estimation of real-time traffic load conditions trying to cope with the variability of network conditions, which is an inherent characteristic of WMNs. The common characteristic of the works referenced above is that they rely only on simulation-based evaluation of the proposed mechanisms.

Recent research studies in the field of WMNs jointly consider problems that traditionally were considered in isolation, such as association and routing. However, as simulation models [30] are not not able to capture the interaction among different layers [31], research related to WMNs is mainly performed in experimental facilities. A recent work in the field [37] proposes a cross-layer association mechanism, which is implemented and evaluated through experimentation in a wireless testbed. However, the evaluation of the implemented scheme is restricted in experiments conducted in a small scale testbed composed of conventional laptop computers and not in a customized large scale Mesh testbed.

At this point, we argue that approaches proposed for WMNs should be fully implemented and properly evaluated through extensive experimentation under real interference and congestion conditions. In an effort to support realistic and large-scale experimentation with heterogeneous network platforms, both the GENI initiative in the U.S. [38], as well as the FIRE initiative in Europe [39] are currently investigating federation of heterogeneous testbeds.

3.2.2 *Integration of Heterogeneous Experimental Facilities*

An initial effort on federation of testbeds was proposed in [40], where the wireless EmuLab testbed and the wired planetary-scale PlanetLab testbed [41] were integrated through the *EmuLab-PlanetLab portal*. The integrated interface provided useful extensions to the PlanetLab's management system. Moreover, several integration challenges were identified for the first time and appropriate solutions were provided. Another work, proposed in [42], aimed at integrating PlanetLab with the ORBIT wireless testbed. The authors considered also the ability of performing experiments on the integrated framework concurrently. Although the PlanetLab testbed provided support for virtualization of resources in the wired part, virtualization of the wireless part had to be further investigated in order to overcome the issues that the broadcast nature of the wireless medium generates. Two discrete integration models were proposed in this work, where the first one aimed to support PlanetLab users in extending their experimental topologies with wireless nodes, while the second one was introduced to provide users of the ORBIT testbed with the extra ability of adding wired network extensions to their experiments.

An important issue that the aforementioned federation approaches had to cope with was the scarcity of a common management system, as well as a common experiment description language. However, this issue was overcome with the introduction of OMF, which provides tools for the management and execution of experiments on testbed infrastructures. Nowadays, OMF has been deployed and used on multiple testbeds supporting many different types of technologies. The work proposed in [43] presented the integration of an OMF-controllable WiFi testbed and PLE, through the addition of an extra wireless interface in PLE nodes that were located within the range of the wireless testbed. This integration was achieved through the development of special tools that supported the definition of slice-specific routing table rules and the exclusive use of the wireless testbed by a single experimenter. Although this integration attempt provided an integrated environment, where all resources could be instrumented through OMF, it also faced the drawback of realizing the wireless testbed as a single resource and thus

limited the access of the federated environment to a single user for each reservation slot.

In an effort to maximize the utilization of OMF-based experimental facilities, NITOS introduced a testbed Scheduler [44] that enables the assignment of different subsets of nodes and channels to different users during specific reservation slots. The work in [45] proposes an integration architecture, which combines an OMF-based wireless testbed supported by the NITOS scheduling mechanism with several PLE OMF-enabled nodes. The resulting federated environment formed a realistic global-scale WMN that supported the execution of multiple concurrent experiments, through the NITOS Scheduler. Moreover, the authors demonstrated an experimental scenario that provided interesting insights regarding real-world experimentation with peer-to-peer systems.

3.3 ONELAB FEDERATION OF NITOS AND PLANETLAB

OneLab [46] is an initiative to provide an open, general-purpose experimental facility, aimed at promoting innovation among network and ICT researchers in Europe, both in academia and industry. It is primarily based on the results of two EC FP7 projects, namely the Onelab and Onelab2 projects. One of the most important goals of the initiative is to establish a federated environment between different, possibly heterogeneous testbeds. As several testbeds have been deployed independently by research institutions across Europe during recent years, and each of them has developed or adopted a different control and management framework, a complex and inconvenient mosaic arises. In this mosaic, experimentation in different testbeds implies familiarization with the respective control frameworks, while combined experiments between different facilities are extremely difficult to setup. The federation between NITOS and PLE, two testbeds of entirely different architecture, which took place during the Onelab2 project, demonstrated that through agreements and collaborations among the involved administrative entities, it is possible to establish architectural paradigms that allow for combined experiments across heterogeneous platforms. In this Section, after describing the two facilities, we analyze the components of the federated environment, which allowed for a combined experiment.

3.3.1 *PlanetLab Europe*

PlanetLab Europe is the European portion of the publicly available PlanetLab testbed, a global facility for the deployment of new network services. It is tightly federated with PlanetLab Central, offering a total of 1000+ nodes worldwide. Each node runs a set of

isolated virtual machines that run either internal PlanetLab services (like e.g. for accountability purposes), or widely and publicly available services (like e.g. CoDeen for content distribution), or more narrow-term experiments that focus on a specific networking environment. PlanetLab has been a pioneering system with respect to federation. PlanetLab Central and PlanetLab Europe have been running the facility in a decentralized manner for more than 6 years now, even if this relies on a PlanetLab-specific paradigm, that is currently being replaced by the much more general Slice-based Facility Architecture (SFA) paradigm [47].

3.3.1.1 *PlanetLab Europe capabilities*

In addition to regular infrastructure servers, PlanetLab Europe offers a few unique features, including:

- reservable nodes: PLE exposes a few of its nodes as reservable, which means that these nodes only run one user slice at any given time. As a result, the experimenter needs to reserve the nodes at specific timeslots, except from adding the corresponding slice to the reservable node, so that the corresponding virtual machine shows up accordingly.

- topology management: PLE also provides basic tools for implementing overlay topologies based on tunnels, either inside a PlanetLab slice or towards the outside world, wherever a tunnel can be terminated.

- wireless nodes featuring WiFi capability; this can come in handy when setting up an experiment that involves end users connected through wireless links; for more focused studies of WiFi, dedicated testbeds are of course more relevant.

- an embedded emulation tool, named *Dummynet* [48], which is able to perform basic actions like packet dropping or delaying and also embeds various predefined models so as to treat a link like e.g. a Wifi link.

3.3.1.2 *Experimental Plane Tools*

Historically, PlanetLab was known to not provide many features in terms of experiment control; given that every virtual machine is accessible through *ssh*, it is of course possible to resort to wide-purpose ssh-based tools like *pssh*. Over the recent years, more tools have been added, for much more powerful control of experiments. Among these,

let us quote the OMF Experimental Controller tool, which enables experiments to be entirely controlled through the OMF experiment control framework. Another example of an experiment management tool is *NEPI* [49], which has the ambition to support a wide range of hybrid simulated/emulated/real experiments, based on the use of a custom developed PlanetLab driver. As an example, creating topologies within PlanetLab is definitely much simpler with *NEPI* than it would be by using the basic PlanetLab tools.

3.3.1.3 *Slices/Slivers*

The notion of a slice is a rather central notion in PlanetLab; it typically allows the modelling of resource allocation, by relating a set of users and a set of resources (nodes). Once created, the slice "owns" one private server (sliver) on each of the selected nodes, and to the designated users, being part of the slice means UNIX shell access to all these slivers. The PlanetLab software is tailored for smoothly orchestrating a complex workflow that involves a large number of people, with different roles (from the legal paperwork, down to locally vouching for users and remote IT management); it also needs to deal with accountability of the resulting network traffic, especially given its scale and diversity of usages, that by design often leads to untypical shapes of traffic; but it admittedly offers little help in managing a slice, and encourages users to leverage third-party tools for the actual experimentation phase.

3.3.1.4 *MyPLC*

MyPLC [50] is the software that was packaged by the PlanetLab operators to let others run their own private PlanetLab system. It was created by Princeton University and is currently being codeveloped by Princeton and OneLab partner INRIA. It provides a ready-to-install set of packages, for both infrastucture-side (XMLRPC API, with related database, software server for securely booting and upgrading nodes), and node-side (slivers management, accountability, remote operations and monitoring). MyPLC is rather flexible, and several tens of instances of MyPLC have been deployed around the world, either for entirely local testbeds, or at the scale of a research consortium.

3.3.2 *NITOS testbed*

NITOS is a wireless testbed featuring 50 WiFi-enabled nodes that are deployed outdoors in the premises of a University of Thessaly campus building. It is remotely and

publicly accessible to any researcher wishing to use its resources, after a plain registration procedure and its approval by the testbed administrators. Below, we describe the two basic software entities of the NITOS testbed, OMF/OML and NITOS Scheduler.

3.3.2.1 *OMF/OML*

NITOS has adopted OMF as its testbed control and management framework. The architecture of OMF is based on three main software components: the Aggregate Manager (AM), the Experiment Controller (EC) and the Resource Controller (RC). The AM provides a set of services to the testbed (inventory, image loading, etc.). The EC, which is the user's interface, receives and parses an experiment script describing configuration of resources and the actual experimental scenario. This script is written in a domain-specific language called OEDL (OMF Experiment Description Language). The instructions in the script are transferred to the RCs of the respective resources, which are responsible to perform the local configurations and application invocations. The different components communicate asynchronously through an XMPP publish-subscribe system, where each message is transferred to an XMPP server, which relays it to its intended destination.

OML (OMF Measurement Library [51]), a companion framework for OMF, is responsible for handling measurements. It consists of two architectural components, the OML server and the OML client libraries. The client libraries are responsible for capturing measurements generated at the resources and, possibly after some manipulations, injecting them in streams headed towards the OML server. The OML server receives the data and stores them in organized databases, one per experiment. The measurements are automatically timestamped; thus it is straightforward to plot them against time. An extremely user-friendly web-based graph application is also associated with OML. The user can point his browser to a specific address and port, and be able to see the evolution of the variables being measured at real time. The exact set of variables to be monitored is specified in the OMF experiment description file.

The typical requirement for the user when using OMF/OML is to develop OMF/OML wrappers for the applications that will run on the resources during the experiment. The role of these wrappers is two-fold. First, they act as proxies translating shell commands to a Ruby-based format that can be parsed by OMF. Second, specific output variables of the source application are linked with the OML client library, so that their values can be sent to the OML server during the experiment execution. Wrappers for several popular network testing applications have already been developed and are typically available at testbeds featuring OMF support. In cases where a user is

NITOS XMPP PubSub records
"/OMF/efkerani"
"/OMF/efkerani/resources"
"/OMF/efkerani/resources/omf.nitos.node023"
"/OMF/efkerani/resources/omf.nitos.node022"
"/OMF/efkerani/resources/omf.nitos.node019"
"/OMF/efkerani/resources/planetlab.test.upmc.ple2"

Table 3: Instance of NITOS XMPP PubSub nodes structure

not satisfied by this set of applications and wants to develop additional wrappers, the procedure is very simple, and it is documented in detail on the OMF website [52].

3.3.2.2 *NITOS Scheduler*

NITlab has developed a reservation and access control software tool for NITOS, called NITOS Scheduler. This tool provides a web-based reservation front end for users of the testbed, accessed through the NITOS website, and a set of back end scripts, daemons and services responsible for controlling access to the testbed's resources, according to the corresponding reservations. In the front end, the user reserves a set of nodes and frequencies for a time duration not exceeding four hours, with a half-hour granularity. In the Scheduler's back end, two main functionalities are worth mentioning: the Scheduler's interaction with the XMPP framework used in OMF and the Spectrum Slicing framework. Both of them are utilized to enable slicing of the testbed, that is, to enable simultaneous experimentation by multiple users through allocation of disjoint sets of resources. Each user at NITOS is associated with a slice. Unlike the typical PlanetLab setup, where slices imply the existence of virtual machines, at NITOS a slice is an abstract entity. In Table 3 we can see a snapshot of part of the NITOS XMPP PubSub nodes tree-like structure, as observed at a given time instant. For each slice in NITOS, the PubSub nodes /OMF/<slicename> and /OMF/<slicename>/resources are always present. When a reservation for a NITOS resource starts, an additional PubSub node /OMF/<slicename>/ resources/ <resource_name> is created. When this reservation ends, this entry is deleted. As a result, access to NITOS nodes is restricted through this dynamic association and disassociation of resources to the corresponding slices, based on queries to the NITOS scheduler's database. Since all OMF communication takes place via the XMPP protocol [53], this mechanism is equivalent to OMF-based dynamic access control of the NITOS resources. Indeed, when the PubSub node

/OMF/<slicename>/resources/ ResourceID is not present, users of slice <slicename> cannot access the RC of resource ResourceID and, therefore, they cannot send OMF commands to it. The addition and removal of PubSub entries is controlled by a back-end daemon, running every half an hour, so that the reservation data as stored in the Scheduler's database exactly matches the granted access rights.

Regarding spectrum slicing, when users submit their reservation, they are only allowed to reserve frequencies that are orthogonal to the ones selected by other users, who have made a reservation for the same or an overlapping time interval (802.11a frequencies are also available, so that sufficient flexibility is achieved). The Resource Controller software provided by NITOS has been modified to allow only frequency configuration requests that are compliant with the existing frequency reservation data in the scheduler database. In this way, cross-interference among simultaneous OMF experiments is avoided.

3.3.3 *Federation Framework*

In this subsection, we describe the basic components of the federated environment between NITOS and PLE. The development of these components took place during the project Onelab2 and enabled, from an architectural point of view, the conduction of the experiment presented in this paper.

3.3.3.1 *Single Sign Up*

One important characteristic of a federated environment is that a user of such a facility should not be obliged to register at its different components separately, but instead be able to use common credentials. To achieve this operation between NITOS and PLE, a single-sign up mechanism has been developed, so that any user of PLE can log into the NITOS portal without going through any extra registration process. This single-sign up process is based on PlanetLab's standard XMLRPC user authentication API. In particular, when a user attempts to log into the NITOS portal, providing a username and a password, the portal's underlying code not only tries to match the credentials with an entry from the native user database, but also contacts the authorization server of PLE through the standard API. If a match is found among PLE's users, an affirmation is sent back to NITOS, which then automatically generates (in the case that id does not already exist) a slice in NITOS having the name of the PLE slice and moreover logs him in the NITOS portal with the provided credentials. The process is transparent to the user and incurs no significant delay. An improved procedure of single sign up

will be offered through the SFA federation framework, making use of the SFA Registry service [54]. This is a coherent and scalable solution. It is gradually being incorporated in several testbeds in Europe, and it is expected to replace earlier schemes, such as the aforementioned solution. This service will be accessed with an SFA client application. Currently, the most advanced SFA client available is MySlice [55]. This graphical client application will also provide the capability to reserve resources from multiple federated testbeds, along with other attractive features. MySlice can already handle reservations of PLE nodes, while an enhanced version of NITOS Scheduler will be incorporated in this tool soon.

3.3.3.2 *Deployment of OMF/OML at PLE resources*

A major difficulty when trying to run combined experiments using heterogeneous facilities is that different languages are used to describe resource configurations and actions. There is the need for agreement to use a common language for experiment description (ED), which must be able to handle the broadest range of resource types possible and easily add support for new resource types in a modular fashion. OEDL is a perfect candidate, as it meets these requirements.

Therefore, PLE decided to incorporate OMF support on demand, in the form of so-called 'OMF-friendly' slices. For slices with this tag activated, at slice-creation time, a pre-installed and pre-configured OMF Resource Controller is initiated in the related slivers. In this way, a PLE resource can be viewed as any other resource of an OMF-based testbed and it can be configured through instructions issued by the experimenter in an experiment script written in OEDL.

3.3.3.3 *XMPP Communication using slices*

OMF employs a Pub-Sub asynchronous communication scheme between the EC and the RCs running on the resources, based on the XMPP protocol. In order for all the resources to be able to communicate with the EC, they must be registered in the same XMPP server or in a set of XMPP servers peered with each other.

In the case that the first approach is followed, nodes belonging to the remote testbed should feature public IPs addresses, so that efficient communication with the single XMPP server is feasible. This works in the case of PLE nodes, but not for NITOS nodes that use private IP addresses. By following the first approach, experiments can only be initiated by NITOS nodes, while users have to configure RCs of the local testbed re-

Figure 9: XMPP server to server peering in OMF

sources to enable communication with the remote XMPP server. In order to provide for a smoother node communication and measurement collection procedure, we adopted the second option, where each resource talks to its local XMPP server, and messages are transferred to their final destination via the XMPP server-to-server (S2S) protocol. The added value is that experimenters do not have to proceed with configurations of the local testbed RCs,and thus the overall procedure becomes totally transparent to the end-user. Therefore, the NITOS and PLE testbeds enabled incoming server-to-server connections in their local XMPP servers, by performing some relatively simple configuration tasks.

In a federated OMF experiment, the PubSub nodes related to a single slice all reside on one of the XMPP servers, which acts as the "rendezvous point" for all participants in the slice. Messages published for that node by users connected to other XMPP servers are relayed to that host via the server-to-server protocol. This architecture is depicted in Fig. 9. The user can freely decide about where the corresponding slice will be hosted. In the case of the federated experiment described in this paper, we only had two XMPP servers, and we decided to host the slice PubSub nodes in the NITOS server.

When this communication procedure is followed, the Resource Controllers of the nodes to which the messages have to be relayed using the S2S protocol, must be configured to use the same XMPP domain, which corresponds to the server hosting the slice's PubSub nodes. In particular, in the RC configuration files of the PLE resources used in our experiment, we had to set the parameter *xmpp.domain* to the value *nitlab.inf.uth.gr*, which is the NITOS XMPP server that hosts the experiment's slice.

It must be noted here that OEDL scripts are completely independent and ignorant of slices. In an experiment description, each resource is referred to by its unique HRN, e.g. *omf.nitos.node019*. In this way, experiment description scripts can easily be reused by different slices. In Section 1 of the Appendix, we present a code snippet example that shows how resource names are used in the OEDL script of our experiment.

3.3.3.4 *Advantages of Experimenting in the Federated Enironment*

The use of the aforementioned tools provides the user with a number of benefits, compared to the case where he attempts to conduct an experiment combining resources from the two testbeds without the existence of federation tools. Below we summarize these benefits, organized according to the expected workflow associated with the user experience.

No Federation:

- *User Authorization*: The user needs to create and maintain one account per testbed
- *Resource Reservation*: The user needs to reserve resources from each testbed separately, potentially through different tools
- *Experiment Setup and Orchestration*: Considerable expertise in network programming and extensive effort is needed to coordinate distributed actions. Moreover, potentially existing and incompatible frameworks are likely to create obstacles in coordination.
- *Measurement Collection*: Incompatibilities in format of measurements lead to extensive post-processing. Extensive scaffolding development work is required to collect and synchronize measurements.

NITOS-PLE Federation:

- *User Authorization*: User only needs a single account
- *Resource Reservation*: No federation in this aspect was present at the time of the experiment presented in this paper. However, the MySlice SFA client tool, which already works with PLE and is being incorporated into NITOS, will soon provide the capability of multi-testbed resource reservation from a single point.
- *Experiment Setup and Orchestration*: OMF does all the background orchestration work for the user. Moreover, a single experiment description file using a single common language is used for description of the distributed experiment, leaving no room for inconsistencies.
- *Measurement Collection*: Common format for measurements, directly parsable even at real time, removes the need for post processing of measurements. Trivial development effort is required for inserting custom measurement points into applications.

3.4 PROPOSED ASSOCIATION MECHANISM

In order to demonstrate the usefulness of the federated environment that combines the wired PLE with the wireless NITOS testbeds, we developed a novel association mechanism proposed for WMNs that is end-to-end performance aware. In this Section, we describe the developed association mechanism and moreover provide details about its driver level implementation.

3.4.1 *System Model And Metrics Definition*

As end-to-end performance in WMNs depends also on the performance experienced on the wireless backhaul part of the network, as well as on the wired infrastructure on which the IGWs are connected, both factors are taken into account by the proposed mechanism to provide for efficient associations. In this work, we consider a special case of WMNs that do not feature a wireless backhaul part, but are composed of MAPs that are directly connected to the wired infrastructure and thus operate as IGWs. A representation of the described topology is illustrated in Fig. 10.

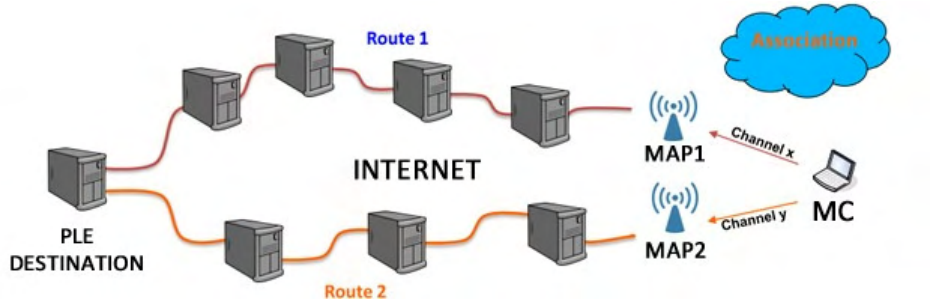


Figure 10: Topology representation

Each MC chooses to associate with a single MAP among the MAPs that operate in its vicinity. Each network node n has a set of neighbors, that reside in its sensing area and operate on the same channel with n. This set of "1-hop" neighbors, that can be either MAPs or MCs, is denoted by A_n. In our previous work [32], we decided on two discrete throughput based metrics for uplink and downlink communications that conform with the special case of infrastructure 802.11 networks. In this work, we consider only the case of uplink communications, which provides for a simple analysis of the proposed mechanism. In uplink communications, frames are transmitted by each MC and destined to the specific AP it is associated with. In our previous work, we arrived at an expression that considers the medium sharing of each MC_i with its "1-hop" neighbors (A_i) and estimates throughput on uplink as follows:

$$T_{ij}^{up} = \frac{1}{\frac{f_i}{R_{ij}} + \sum_{k=1}^{|A_i|} \frac{f_k}{R_k}}, \tag{8}$$

where R_{ij} and R_k denote the PHY rates used by MC_i and each node $k \in A_i$ accordingly, while f_i and f_k are defined as activity indicator factors reflecting the activity intensity of MC_i and node $k \in A_i$ in comparison with each other.

3.4.2 *End-to-end performance aware Association Mechanism*

Based on the analysis in the previous Section, we are able to estimate throughput performance for the single-hop access link between the MC and each potential neighboring MAP. More specifically, the denominator of expression 1 estimates the average time duration required for a single bit of information to be transmitted over the access link. In this work, we develop an association framework that is based on Round-Trip Time (RTT) measurements. Readers interested in related work including comparative study of more sophisticated association metrics are directed to our previous work [32]. In this paper, we estimate the RTT required for the initial transmission and subsequent retransmissions of a frame with specific length, by multiplying the calculated delay with the number of bits that are transmitted over the access link and moreover double the resulting value to estimate the total delay required for both transmissions. Concluding, we estimate the RTT for a specific frame of M bits that is transmitted over the access link from MC_i to MAP_j and back again, as follows:

$$RTT_{ij}^{A} = 2 * M * (\frac{f_i}{R_{ij}} + \sum_{k=1}^{|A_i|} \frac{f_k}{R_k}) \tag{9}$$

In our approach, we develop a simple mechanism to estimate RTT experienced on the wired backhaul part of the network as well. More specifically, each MAP_j periodically transmits probe packets and measures RTT_j^B for the wired network backhaul. These values are broadcasted to all MCs in range and as a result each MC_i is able to estimate end-to-end RTT for each potential MAP_j, as follows:

$$RTT_{ij}^{total} = RTT_{ij}^{A} + RTT_j^B \tag{10}$$

3.4.3 *Implementation Details*

For the implementation of our mechanism, we used the Mad-WiFi open source driver. Details about the mechanism aiding in performance estimation on the wireless part can be found in our previous work [32]. In this Section we will provide details about the developed mechanism that enables application layer information regarding wired RTT information to reach neighboring MCs. First of all, we use a simple application level program that runs at the APs and sends probe packets to the destination host to calculate RTT_j^B values. In order to broadcast this information to all neighboring MCs, we first had to make the RTT_j^B value available to the kernel level, as all MAC layer mechanisms are implemented as loadable kernel modules by the MAD-WiFi driver. An efficient way to transfer information to the kernel is through the proc virtual filesystem, which resides in the kernel memory. The proc files used by the MAD-WiFi driver are stored in /proc/sys/net/wlan/athX, where X denotes the specific interface. Another script running locally at the APs periodically writes values to the specified proc file and as soon as a new record is written the driver is informed. As for the next step we had to broadcast the RTT_j^B value to all neighboring MCs. In order to do this, 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 [1]. The MCs constantly estimate the RTT_{ij}^A values for each potential MAP. The third step is also performed at the MC side, where the driver combines the RTT_j^B value with the RTT_{ij}^A and calculates the RTT_{ij}^{total}. Finally, each MC_i associates with the MAP_j that features the lowest RTT_{ij}^{total} value.

3.5 EXPERIMENTAL SETUP

In this Section, we provide details about the measurement methodology and the various tools that we used in order to establish a proper experimental setup. Moreover, we provide a brief description of the OEDL ED script that we used for the orchestration of the conducted experiments.

3.5.1 *Measurement Methodology*

Fig. 11 represents the actual topology used in our experiments. We consider a typical scenario, where one traffic flow is generated from the MC node and relayed through the two available MAPs to the final PLE destination node. The MAPs act as IGWs and get

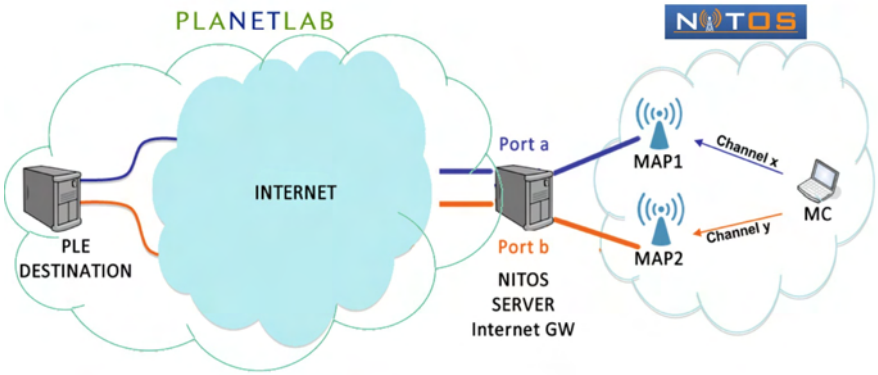


Figure 11: Experimental topology representation

access to the wired network part through NITOS Server. As NITOS nodes are assigned private IP addresses, we had to enable a Network Address Translation (NAT) service at NITOS Server through proper IPtables [56] configurations, in order to provide Internet access to the two nodes operating as MAPs. We also followed a similar procedure to provide for proper relaying of traffic generated by the MC through the two MAPs.

In a part of our experiments, we required the injection of artificial delay in the wired backhaul link. For this purpose, we used the *Dummynet* [48] tool, which is able to emulate queue and bandwidth limitations, delays, packet losses, and multi-path effects, by intercepting packets in their way through the protocol stack. As an outcome of the OneLab project, PLE natively supports *Dummynet* as a kernel module in all nodes, configurable from the sliver through a command-line tool. Since it is straightforward to wrap shell commands inside the OMF experiment description file, we used this approach to avoid having to log into the remote PLE node and issue the commands manually. Furthermore, incorporating the *Dummynet* commands inside the OMF script allows perfect synchronization with the rest of the applications and with the measurement collection process.

As all packets received at the destination, share the same IP address of the NITOS Server, we base packet discrimination on the port numbers. To this aim, we used a simple *Nmap* [57] script to dynamically detect the specific ports used for incoming connections at the PLE node. For the wireless part, we enable a pair of nodes that operate on each of the channels used by the MAPs and generate channel contention conditions of varying traffic rate. Fig. 12 shows a screenshot of the OMF visualization tool representing delay emulation for two discrete flows generated by each one of the MAPs, as monitored at the PLE node. The throughput performance of the experiments is measured by using Iperf [58]. In our experiments, we run an Iperf Client at the MC to generate TCP flows and UDP flows of varying rate and also an Iperf Server residing at the PLE node to receive traffic and collect the corresponding measurements.

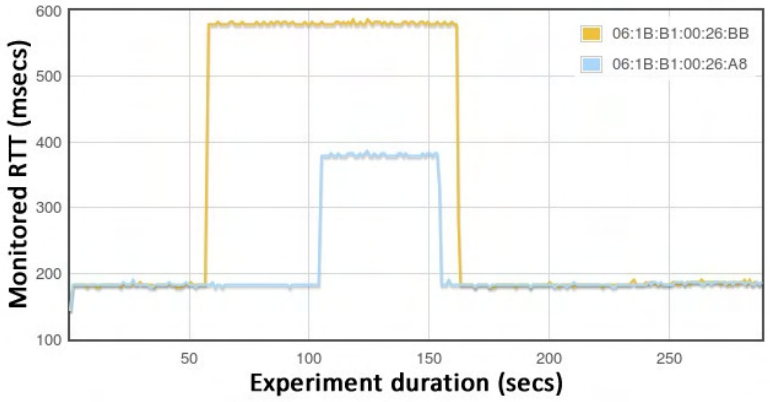


Figure 12: RTT monitored at the PLE for two flows generated by each MAP

Considering the video experiments presented last in Section 6, we used the *Video LAN Player (VLC)* [59] to stream a video of H.264 format from the MC to multiple PLE destination nodes. In order to provide for a simple experimental setup, we run both the VLC Server and Client applications on the MC node and also utilized the *Socat* [60] platform to be able to relay, through UDP sockets, all packets that are received on a specific port at the PLE node back to the MC.

3.5.2 *Experiment Description*

We described our experiment with the OMF Experiment Description Language (OEDL) script. An OEDL script is basically comprised of two parts. The resource configuration part and the experimental scenario part. In the first part, configuration parameters for both the network interfaces of the resources and the applications to be executed on them are determined, while the measurements to be collected are also defined. A brief code snippet from our ED, which describes the configuration of the first MAP, is presented in Section 2 of the Appendix.

We place a single resource in the group called 'MAP1', determine two applications to run on this resource (OMF wrappers for these applications must be present in the resource's image), and we also define values for the properties of the wireless interface to be brought up and used. For the purposes of the experiment, we used some existing OMF wrappers for common applications, but also developed some new ones for the specific needs of our experiment. For instance, we developed a wrapper for the *iwconfig* application, which is used to dynamically identify the current association of the MC, for visualization purposes.

In the second part of the ED script, the exact course of actions to take place is determined. The timing of the different tasks is described, i.e. when to start the applications on a resource, how long they will run, or when they will stop. Moreover, we are able to define the exact time that specific events will take place, e.g. the injection of artificial

delay that is executed through the call to the *Dummynet* application. Considering the *Dummynet* application, we did not develop a custom wrapper, as there is no need for gathering any measurements from this application. The *Dummynet* application is called using the exec command, which in this case is applied to the group of the PLE node, using standard delays defined at the beginning of the experiment. Based on the custom developed *Nmap* script, we are able to dynamically detect the specific ports used for incoming connections at the PLE node and we feed the output of this script as an input parameter to the *Dummynet* application, which then injects delay solely to flows that use the specified ports. Representative code snippets that are used to configure both the delay attribute that are used by the *Dummynet* applications, as well as to instruct the *Dummynet* application, are presented in Section 3 of the Appendix accordingly.

The RTT measurement applications are Ruby applications, that use the client-server model. The client side is installed at the MAPs and the server side is installed at the PLE node. Since the applications we used are written in Ruby, we used the OML library, in order to inject the measurements gathered during runtime i.e. RTT values, to the sqlite experiment specific database that is created by the OMF. More details regarding the experimental description, as well as all the important applications and scripts are publicly available and can be downloaded directly from the NITOS web portal [61].

3.6 EXPERIMENTAL EVALUATION

In this Section, we evaluate several experimental scenarios that aim at presenting the abilities offered by the federated testbed environment, rather than presenting innovative research results. More specifically, these experiments have been designed to demonstrate the effectiveness of association mechanisms that jointly consider factors affecting both wireless and wired performance. The execution of such combined experiments requires integrated testing, which would not be feasible without the existence of the federated environment.

The conducted experiments are organized in two parts, where in the first part we measure the performance for a static topology, where the MC communicates with a single MAP that is forwarding traffic to a single PLE destination. Under this topology, we focus on measuring the resulting performance under various settings of artificially injected delay in the wired part and channel contention in the wireless access link as well. In this first part of experiments, we decided to vary the wired delay artificially and not by using different network paths, in order to isolate the impact of other factors, such as traffic congestion that would significantly vary among different network topologies.

In the second part of our experiments, we do not alter any settings in the wireless network part, but we focus on testing the performance of the proposed scheme under realistic RTT variations in the wired part, which conditions are generated through the use of different PLE nodes as the final path destinations. In order to accomplish this, we use multiple PLE nodes that span across different continents and moreover offer different load conditions, thus featuring diverse end-to-end characteristics for the wired part.

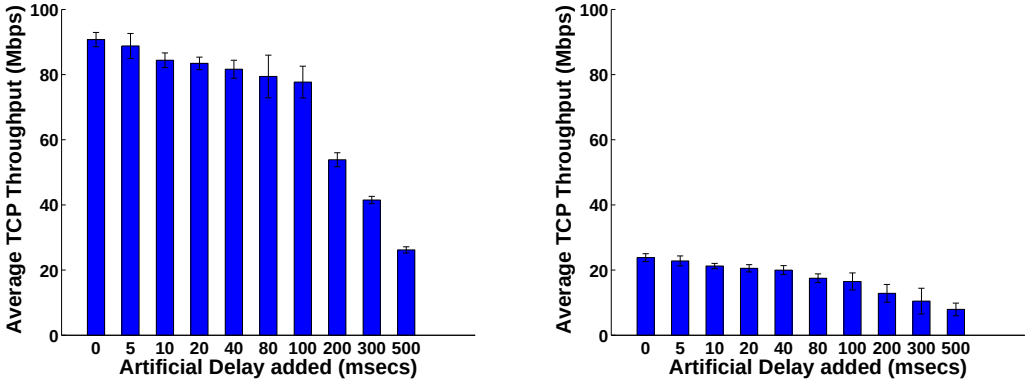
3.6.1 *Experimentation based on a single PLE node*

The experiments described in this first part are organized in two sets, where in the first set we generate conditions of varying delay in the wired backhaul part, while in the second set we vary the delay in the single hop wireless access link. Moreover, each experiment is performed in two discrete phases, where in the first one we compare the effect of injected delay on performance affecting either the wired (1st set) or wireless (2nd set) part solely, while in the second phase we consider the impact on the combined network architecture.

The conducted experiments aim at presenting the performance improvement that can be offered through the application of the proposed association mechanism and thus measure the performance for a static scenario, where the MC communicates with a specific MAP. Under this scenario we alter the delay induced in each part and monitor the resulting performance in terms of TCP /UDP throughput, packet loss and jitter values. The initial RTT in the wired part between NITOS Server and the PLE node that is used in these experiments and resides in France is around 80 ms, while in the wireless part the reported RTT between the MC and each MAP without any external contention is below the value of 1 ms. In all the conducted experiments, the default Rate adaptation algorithm of the driver has been used.

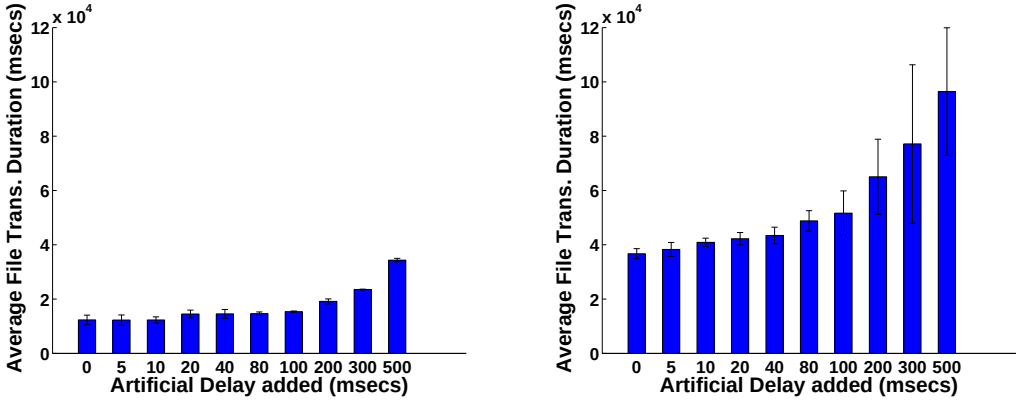
3.6.1.1 *Wired - Combined set of Experiments*

In this first set of experiments we use the *Dummynet* tool to generate artificial delay. Fig. 13a and Fig. 13b illustrate the TCP throughput achieved under various artificial delay values in the wired and the combined architectures accordingly. We notice that even small variation of delay in the wired part significantly affects TCP throughput and therefore should be taken into account. Moreover, we notice that the wireless access link acts as the performance bottleneck that significantly limits yielded performance. A particular observation is that the same experiments provide higher deviation values when conducted in the combined topology, for the cases of 200, 300 and 500 msecs



(a) Wired (b) Combined

Figure 13: TCP Throughput vs Artificial Delay

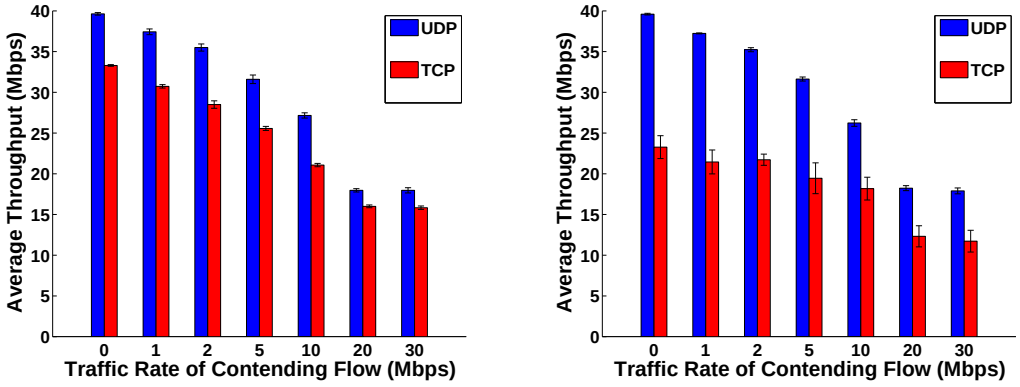


(a) Wired (b) Combined

Figure 14: TCP File Transmission Duration vs Artificial Delay

of injected delay, in comparison with the execution solely in the wired part. However, average throughput values show similar performance in the above cases. Based on the observed results, we remark that relatively high values of injected delay make TCP performance in the combined network highly unstable.

In Fig. 14a and Fig. 14b, we present the duration required for the successful transmission of a file with size of 100 MBs in the wired and combined network accordingly. We easily notice that even a low increase in RTT values of 20 ms increases file transmission duration up to 5,5s (15%). Moreover, we notice that the effect regarding the increased deviation values is also clearly illustrated between Fig. 14a and Fig. 14b. We also conducted experiments based on UDP transmissions. However, UDP performance in terms of throughput, packet loss and jitter is not affected by the artificially injected RTT delay. This comes from the fact that even high values of artificial delay cannot result in packet loss, as the high capacity of operational system buffers supports storage of packets that arrive during the artificial delay interval even at the maximum traffic rate of 90 Mbps that is used in our experiments.



(a) Wireless (b) Combined

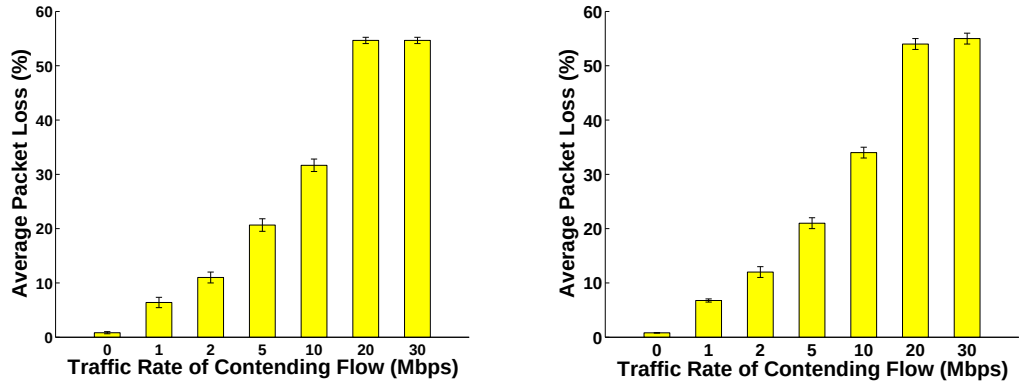
Figure 15: TCP - UDP Throughput vs Contention

3.6.1.2 *Wireless - Combined set of Experiments*

The second set of experiments has been designed to demonstrate the impact on end-to-end performance of channel contention in the wireless access link. Fig. 15a and Fig. 15b illustrate TCP and UDP throughput achieved in the wireless access link and the combined network accordingly, under various values of traffic rate for the contending flow.

For the UDP case, we notice that even contending flows of low traffic rate highly impact performance in both cases. In addition, we observe that results obtained in the wireless and combined networks are very similar and both feature relatively low deviation values. Packet loss measurements illustrated in Fig. 16a and Fig. 16b show that UDP performance is directly related to loss of packets. As the MC injects packets with high traffic rate, the wireless network capacity is exceeded due to the simultaneous transmissions of the contending flow. The resulting channel contention yields packet loss, which cannot be detected by the UDP protocol and thus the rate of data entering the network is not restricted within the network capacity region.

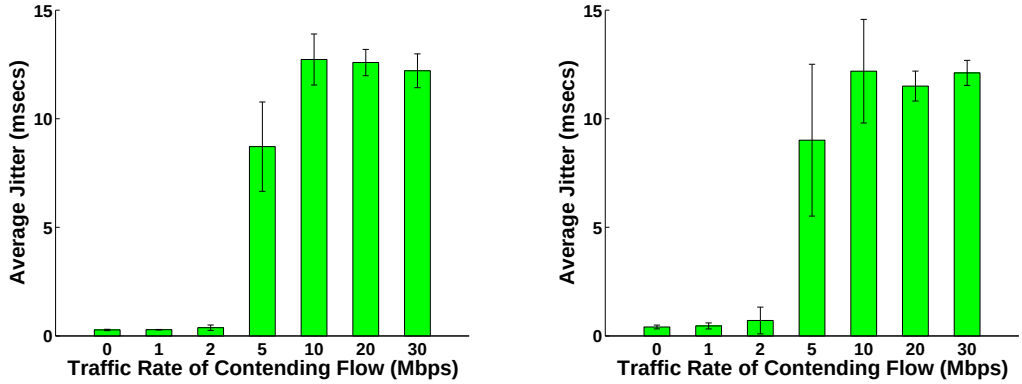
However, in the TCP case, we observe lower throughput performance yielded in the combined network (Fig. 15b). This is due to the fact that the TCP protocol involves RTTs estimation in its adaptive retransmission procedure, as part of the congestion control mechanism that aims at estimating the available network capacity to prevent link congestion. As a result, the TCP protocol reacts upon the detection of increased RTT values that result from the augmented network range and consequently limits the rate of traffic injected by the MC, in order to control congestion in the network. Another particular observation regarding the TCP case is related to the high deviation values observed among the multiple executions of the experiment in the combined architecture, compared with the low deviation values observed in the local wireless network.



(a) Wireless (b) Combined

Figure 16: UDP Packet Loss vs Contention

This is due to the fact that the generated traffic flows go over the Internet through PLE during experimentation in the combined architecture and thus high deviation values are recorded as a regular characteristic of experimentation on realistic planetary scale networks.



(a) Wireless (b) Combined

Figure 17: UDP Jitter vs Contention

Concerning UDP packet loss values presented in Fig. 16a and Fig. 16b, we observe similar performance between experimentation on the wireless link and the combined network. Similar results are also obtained in terms of UDP Jitter between experimentation on the two different network architectures, as illustrated in Fig. 17a and Fig. 17b. Moreover, we used the OMF visualization tool, in Fig. 18, to plot RTT values reported from the two MAPs with red and blue colors and also yellow color for values reported from the MC. Particularly, we observe that the MC is always associated with the MAP that features the lowest RTT delay and moreover we notice a handoff that lasts between the 120 and 140 seconds of the experiment.

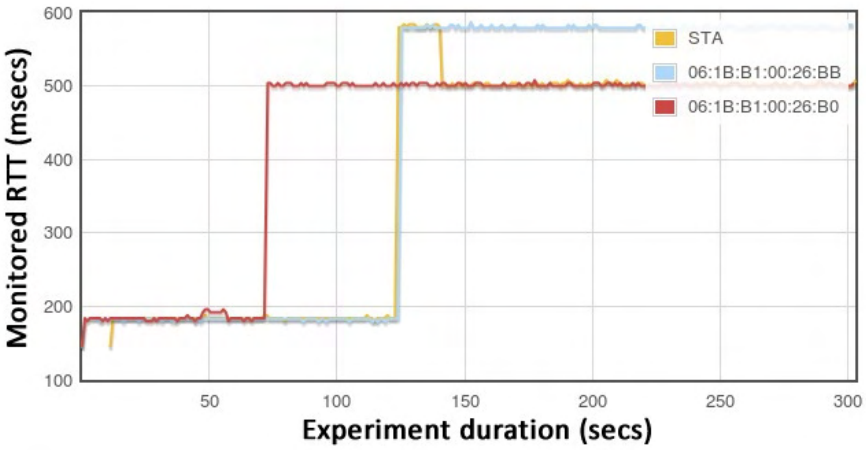


Figure 18: Handoff demonstration

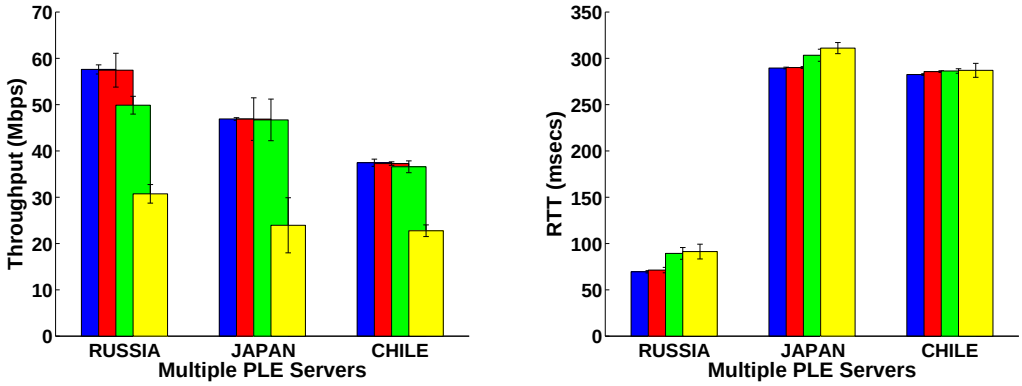
3.6.2 *Experimentation based on multiple PLE nodes*

In this second part of experiments we focus on testing the performance of the proposed scheme under more dynamic topologies, where different PLE nodes are used as the final path destinations. Through the use of multiple PLE nodes that exist in different locations, we are able to evaluate performance under realistically varying RTT values. The experiments described in this part are organized in two discrete sets. In the first set, we investigate the impact of a specific contending flow on the throughput performance of different target flows and under varying levels of traffic contention, while in the second experimental set, we evaluate the performance of a specific flow during the streaming of a video file that is destined towards different PLE destination nodes.

3.6.2.1 *Throughput experiments*

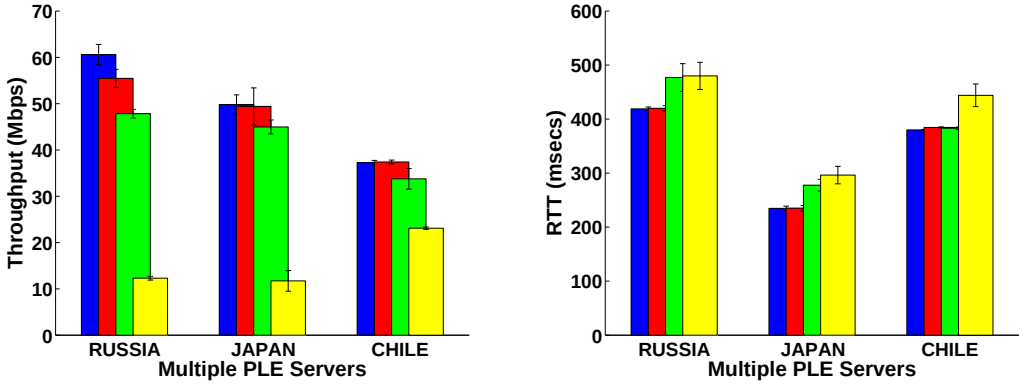
The target flows that are considered in this set of experiments, are originated from different PLE nodes and are destined towards a single PLE node. In order to investigate the impact of traffic congestion on the throughput performance of the target flows, we use a contending flow under various traffic conditions. The contending flow is originated from a node in NITOS and shares the same destination with the target flows. The sources of the different target flows are three PLE nodes that are located in Russia, Japan and Chile accordingly. This experimental scenario has been designed in order to investigate the relation between the measured RTT for the different paths used by the target flows versus the obtained throughput performance in each case.

Fig. 19a and Fig. 19b depict the UDP throughput performance and the monitored RTT values, during experimentation with the different target flows, in the case that the PLE destination node is located in France. Similar results are illustrated in Fig. 20a and Fig. 20b for the case that the destination PLE node is located in Australia. In these



(a) Throughput (b) RTT

Figure 19: Measurements for the target flow that is originated from a node in NITOS and destined to a PLE node in France



(a) Throughput (b) RTT

Figure 20: Measurements for the target flow that is originated from a node in NITOS and destined to a PLE node in Australia

experiments, we start by measuring performance of the target flows, while the contending flow is inactive and continue by altering the level of traffic that is injected by the contending flow among the values of 20, 50, 100 Mbps. The blue bars indicate performance in the case that the contending flow is inactive, while the red, green and yellow bars represent the results obtained in the cases that the contending flow is transmitting at the rate of 20, 50 and 100 Mbps accordingly.

As clearly demonstrated through Fig. 19a and Fig. 20a, the UDP throughput performance of all target flows is significantly affected by the two different contending flows, in the case that the traffic rate of the contending flow reaches the value of 100 Mbps. However, in Fig. 19a, we notice that the target flow that is originated from Russia starts to witness throughput degradation even in the case that the contending flow is transmitting at the rate of 50 Mbps, which is not observed for the other target flows. Moreover, we notice in Fig. 20a that all the different target flows start to witness throughput reduction, as soon as the contending flow reaches the rate of 50 Mbps, except for the

experiment with the PLE node in Russia, which yields degraded performance, even at the rate of 20 Mbps. The observed results can be interpreted, based on the fact that each target flow shares a different part of common network paths with the contending flow and as a result traffic congestion affects their performance in a different degree. On the other hand, the prevailing traffic conditions on each different network path also impact the resulting performance. However, we cannot monitor these conditions that are generated from external to our experiment sources, as the PLE network is a shared environment that uses the common Internet architecture.

Through the RTT values representation, which are illustrated in Fig. 19b and Fig. 20b, we observe that the RTT for all the different network paths increases and also presents higher deviation values, in the cases that the contending flow severely impacts the UDP performance of the target flow. Based on this observation, we remark that variation of RTT values may act as an identification factor of network congestion. We also conducted experiments based on TCP transmissions. However, we omit presenting these results, as the obtained TCP throughput performance, across different network paths, presents the same univocal relation with RTT variations that was observed through the TCP experiments under artificially varying RTT values.

3.6.2.2 *Video experiments*

The goal of this final set of experiments is to show how the federated environment between NITOS and PLE testbeds can be utilized in testing the performance of video streaming applications that utilize discrete network paths. We consider a typical scenario for streaming video, which is based on the topology that is represented in Fig. 11, where a video file is streamed from the MC, forwarded to the MAP it is associated with and finally reaches different PLE destination nodes. The video streamed through the PLE network goes over the Internet giving experimenters the characteristics of a realistic network. Based on the experimental setup that is generated through the use of the *Socat* platform, the MC receives the corresponding traffic streams that have resulted through transmissions across different network paths and is finally able to store different versions of the initially transmitted video sequence. Consequently, we are able to evaluate the perceived video quality in each case, based on comparison of the delivered video files.

The number of received frames across multiple PLE destination nodes that feature different RTT values, is illustrated in Fig. 21. Results show comparable performance for the PLE nodes that are allocated in Greece, Australia, Portugal and Canada. The video file that is streamed between nodes in the internal network of NITOS is delivered

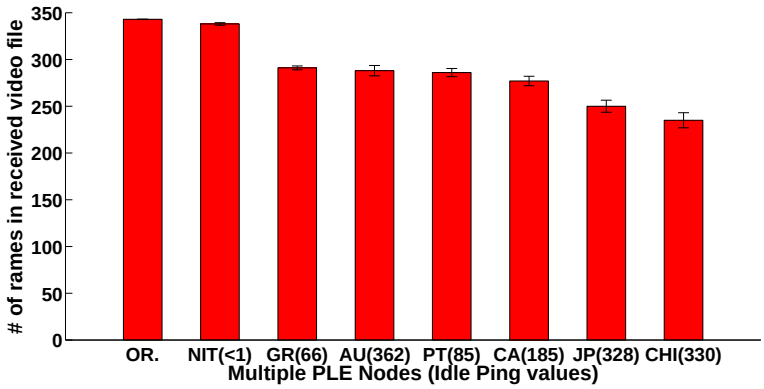


Figure 21: Number of received frames during the transmission of the same video file through different PLE source nodes

(a) NITOS (b) Portugal (c) Chile

Figure 22: Representative screenshots of the same frame as transmitted through different PLE nodes

nearly unvaried. Moreover, we notice that the lowest performance is yielded in the network paths that include PLE nodes located in Japan and Chile. Considering the obtained results, there exists no direct relation between the RTT values and the number of delivered video frames. In addition, we present some indicative screenshots in Fig. 22, which present the same video frame as delivered through different network paths. As our aim is not on comparing the video performance in terms of advanced video quality metrics, we simply rely on the screenshots representation to depict the video quality degradation that is yielded during the streaming of a video sequence through different PLE nodes over the Internet.

3.7 SIMULATION BASED EVALUATION

In order to explore the key differences between realistic testbed and simulated environments in performing Mesh Network experimentation, we decided to reproduce the WMN topology under consideration in a network simulator tool. For this purpose, we decided to use the well established NS3 [62] discrete-event network simulator, which provides for experimentation in both wireless and wired topologies. By properly configuring NS3 to simulate WMN topologies, we can evaluate the comparative complexity

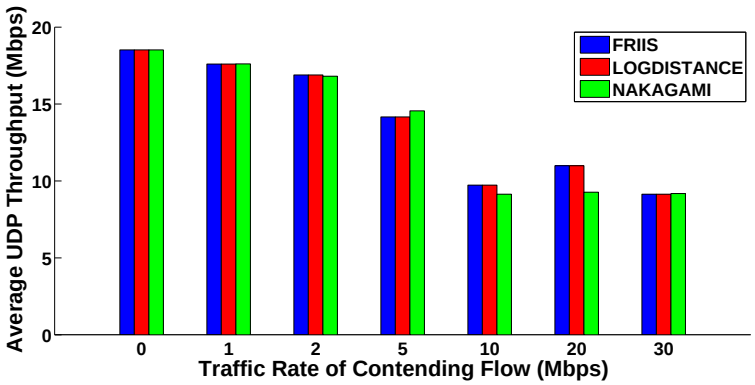


Figure 23: UDP Throughput vs Contention, in a wireless grid topology of 2 meters node distance, considering different NS3 Propagation Models

of configuring simulation and testbed environments. Moreover, we are able to conduct extensive sets of experiments and, through direct comparison between the results obtained in the realistic and simulated environments, we determine whether simulation models are able to reliably estimate the performance of WMNs. To this aim, we reproduced the full set of the experiments presented in Section 6.1, in the NS3 environment. In the rest of this Section, we refrain from showing the full set of obtained results and only present a subset of selected experiments that presented dissimilar performance, with regard to the results obtained in the realistic federated testbed environment. Based on the simulated experiments, we also comment on the obtained results and the perceived user experience.

3.7.1 *Wireless Topology - Channel Contention*

The first testbed experiment that we tried to reproduce is related to the evaluation of the impact that wireless channel contention induces on end-to-end performance of the wireless access link. In order to replicate the actual topology that was used in the real experiments, we simulated a grid topology consisting of 4 wireless nodes that are spaced 2 meters apart. In this scenario, we evaluate the throughput performance of the first communicating pair of nodes, considering varying traffic rate for the second contending flow. The nodes were configured to support the IEEE802.11a standard, through the applied *"WiFi_Phy_Standard_80211a"* tag. We also configured the nodes to use the fixed PHY-bitrate of 54Mbps and 6Mbps, through the application of the *"OfdmRate54Mbps"* and *"OfdmRate6Mbps"* tags, for transmissions of data and control frames accordingly. In order to model the wireless propagation environment, we used the basic Friis [63] model, which assumes that the received power drops as the square of the distance between the transmitter and receiver nodes, as it occurs in free space.

In Fig. 23, we use the blue colour to depict the resulting UDP throughput performance. We easily notice that the maximum achievable throughput of 18.52 Mbps is significantly lower, than the maximum value of 39.6 Mbps that was obtained in the real experiment. Moreover, we observe a remarkable performance anomaly in the case that the contending flow uses the traffic rate of 20Mbps, where throughput increases despite the fact that a higher level of channel contention is configured. In an effort to overcome the observed performance dissimilarities, we decided to experiment with different wireless propagation models, while keeping the node, topology and network configurations stable and exactly corresponding the realistic scenario.

To this aim, we first configured the LogDistance [64] propagation model, which assumes exponential path loss over the distance between transmitter and receiver nodes. More specifically, we set the path loss exponent value equal to 3, which is suitable for modelling urban environments, in an effort to approximate the real environment in which NITOS is located. However, based on the obtained results, which are presented through the red coloured bars in Fig. 23, we notice that performance of the considered topology was not affected by the application of a different propagation model. Our next choice was to increase the node distance, in order to evaluate whether the increased distance would impact the resulting performance under the different propagation models. Nevertheless, although the distance between the two node pairs was increased up to 20m, significantly differing from the original topology, monitored performance in the case of the Friis and LogDistance models was totally unaffected and thus we refrain from presenting the resulting plot.

Our next choice was to apply a signal fading model, and more specifically the Nakagami model [65], which encompasses different fading formulas for short-distance and long-distance communications. The resulting performance is illustrated in the green coloured bars of Fig. 23. Application of the default NS3 Nakagami propagation model implementation resolved the observed performance anomaly in the case that the contending flow was using the traffic rate of 20Mbps, when considering a wireless grid topology with 2 meters of node distance. We also increased the node distance up to 20m and noticed that the Nakagami model presented a slight performance decrease in all the considered cases, in contrast to the Friis and LogDistance models, in which cases performance was totally unaffected by the distance increase.

The overall conclusion drawn through this set of experiments is that several rounds of simulator reconfigurations were required in order to approach the performance obtained in the realistic wireless environment. However, although we configured all the proper settings to exactly resemble the realistic scenario, we did not manage to achieve the maximum throughput performance of 39.6 Mbps that was monitored in the real

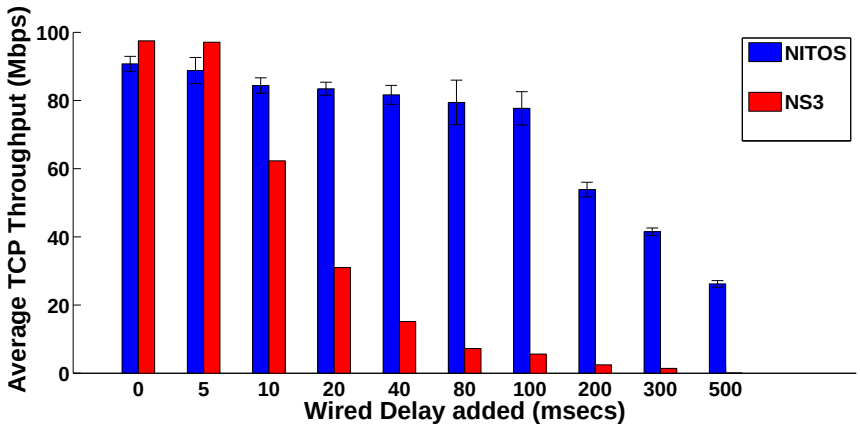


Figure 24: TCP Throughput vs Wired Delay, in wired topology

experiment. Moreover, the obtained results confirm observations of related work [66], where the authors argue that direct application of NS3 propagation loss models with the default parameters, do not result in realistic network behaviour. Based on our own observations, we also remark that adjustment of several NS3 parameters is required to accurately simulate performance of wireless experiments, even when simple scenarios are considered. Another particular observation is related to the fact that there were only minor performance variations observed between measurements collected through multiple executions of the same experiment. As a result, we conclude that NS3 was not able to accurately reproduce the highly dynamic and stochastic nature of the wireless medium in the simulated environment.

3.7.2 *Wired Topology - Injected Delay*

The second testbed experiment that we tried to reproduce aims at assessing the impact of artificially injected delay on end-to-end performance of the wired backhaul part. To this aim, we created a simple topology consisting of two wired nodes that are connected through a point-to-point (P2P) link that offers configuration of the link capacity and delay, through the *"DataRate"* and *"Delay"* attributes. In order to replicate the wired experiment presented in Section 6.1.1, we configured the maximum P2P link capacity equal to 100Mbps and also configured the link delay to vary between zero and 500 msecs. Under this scenario, we monitor the resulting performance in terms of TCP/UDP throughput performance.

In Fig. 24, we illustrate the TCP throughput achieved under various artificial delay values, in both the realistic testbed and NS3 simulated environments. Based on the obtained results, we notice that the NS3 simulated TCP protocol considers RTT estimations as part of the TCP congestion control mechanism, and thus adapts the injected traffic rate, upon the detection of increased RTT values. However, the NS3 simulated

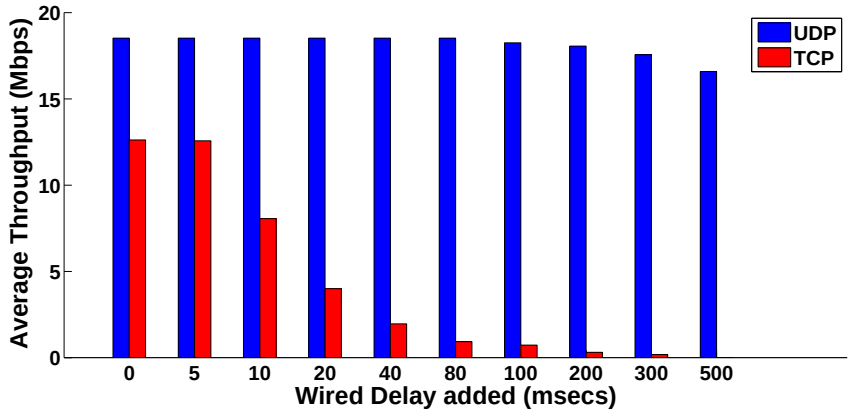


Figure 25: Throughput vs Wired Delay, in combined topology

TCP protocol reacts far more aggressively to the wired delay increase than the TCP implementation that is loaded on the Linux-based operating systems of the PLE and NITOS testbed nodes. More specifically, significant deterioration of the TCP throughput performance is observed, as soon as the delay reaches the value of 10 msecs. Moreover, TCP throughput tends to decrease proportionally to the wired delay until it reaches extremely low values, when delay increases above the 200 msecs. However, in the respective testbed experiments, throughput performance is only slightly decreasing for wired delay value up to 100 msecs, while significant drop, that is proportional to the injected delay, is only monitored for wired delay values above 200 msecs. Similar experiments were also conducted based on UDP transmissions, in which case throughput performance was not affected by the configured P2P delay, thus presenting network behaviour fully identical to the testbed experiments.

Concluding, we remark that configuration of simulated wired experiments required less user intervention and moreover that the performance results approximated to some extent the experimentally obtained ones. In spite of the performance dissimilarities that were observed due to the aggressive behaviour of the TCP protocol implementation in NS3, results on maximum TCP and UDP performance were identical to the experimentally obtained ones. We also observed only minor performance deviation for measurements collected through multiple executions, which shows that the P2P-based delay injection mechanism introduces far less environment variability in comparison to the one implemented through the *Dummynet* tool.

3.7.3 *Combined Topology - Injected Delay*

In the next step, we tried to reproduce the experiment that considers the combined network topology illustrated in Fig. 10. In order to to simulate the wired backhaul between the final server destination and the MAP that provides access to the associated MC, we used a P2P link of configurable delay, as in the aforementioned exclusively

wired scenario. Following an approach identical to the one used in the simulated wired experiments, we aimed at detecting whether the combined topology further impacts end-to-end performance, as it was observed in the respective testbed experiments. For this purpose, we reproduced the combined topology, following settings identical to the ones configured in the two previous simulated experiments, considering each discrete network path accordingly. Under this scenario, we monitor the resulting performance in terms of TCP /UDP throughput performance.

Fig. 25 illustrates both TCP and UDP throughput achieved in the simulated combined network topology, under various wired link delay values. Regarding the TCP throughput performance, we observe that it deviates from the realistic behaviour presented in Fig. 14b. First, we notice that throughput decrease follows a trend similar to the one observed in the simulated wired experiment. Second, we note that the maximum achievable throughput performance of 12.62 Mbps is significantly lower than the one obtained in the realistic experiments (24.1 Mbps). Considering the UDP throughput experiments, we notice that performance is totally unaffected by the injected delay, for delay values lower than 80 msecs. However, in cases when the injected delay increases above 100 msecs, throughputs tends to decrease, reaching a maximum reduction of approximately 2 Mbps, in the case that 500 msecs of delay are configured for the P2P link. UDP performance results obtained in the simulated combined scenario come in contrast with the ones obtained in the testbed environment, where UDP throughput was not affected by the wired delay, even for the highest configured delay values of 500 msecs.

At this point, we would like to present some generic conclusions regarding the experience gained through the execution of identical experiments in both testbed and simulation based environments. Considering the complexity of experimental configuration, we remark that although testbed experimentation requires significant effort in the early configuration stages, the ease of use provided by experiment instrumentation frameworks, such as OMF and NEPI, efficiently simplifies the overall process. On the other hand, execution of introductory scenarios in the NS3 simulator tool was straightforward. However, several rounds of simulator reconfigurations were required in order to approach the performance obtained in the testbed environment, especially when configuring wireless experiments. Regarding the ability of simulator tools to accurately simulate performance of testbed experiments, we reached the conclusion that NS3 is able to provide some basic feedback about the expected performance of real testbed experiments. However, based on the various flaws that were observed, we conclude that NS3 is not able to accurately reproduce testbed experiments, especially topologies that include wireless or combined network parts. We have to mention that our

experience with network simulators is limited to NS3 and that other mode advanced simulator environments, such as OPNET [67], might potentially approximate real experiments more accurately. However, sophisticated network simulator environments require experimenters to obtain high cost licences, in order to have access to the full list of available components. As a result, we decided to use the publicly available NS3 platform that is primarily targeted for research use and directly compare its performance with experimental results obtained through the federated environment consisting of testbeds that provide free remote access to experimenters worldwide.

3.8 CONCLUSIONS AND FUTURE WORK

The unique two-tier architecture introduced by realistic WMNs has directed research efforts towards experimentation on global scale realistic environments that result from federation of heterogeneous networks. In this work, we present the federation of the wired PLE and the wireless NITOS testbeds. The resulting architecture has enabled the execution of realistic association experiments in the context of WMNs, which presented several characteristics of experimentation under real world scale and settings. Several of the testbed experiments were also reproduced in a network simulator tool, demonstrating the inability of simulated environments to accurately approximate the complex nature of the wireless medium and the high variability that experimentation over Internet generates. Through the extensive experiments and the corresponding collected results, we validated the importance of integrating experimental facilities for the design and development of the Future Internet. As part of our future work, we plan on investigating performance of more complex WMN topologies that also feature a wireless multi-hop backhaul in the aforementioned federated environment. Furthermore, we aim at enhancing the existing tools for experimentation in federated testbeds and developing new ones, towards upgrading the perceived user experience.

Part II

SPECTRUM MANAGEMENT

4

DYNAMIC FREQUENCY SELECTION IN WLANS

4.1 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 [23]. Moreover, some vendors, based on traffic measurements, aim to avoid frequencies that are highly congested by nodes belonging to other networks [68]. 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.

4.1.1 *Related Work*

Initial efforts on channel assignments for wireless networks date back to the 1980s [69]. 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 [70], the authors construct an interference graph that requires exchange of information among APs, in order to estimate cell in-

terference. The work in [71] proposes a distributed interference mitigation mechanism restricted in the case of non-overlapping channels only. Another work proposed in [72], 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 [73] 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 [74] incorporate traffic demands of both APs and STAs in the proposed mechanism. Nevertheless, this approach was restricted in considering uniform demands across all associated STAs of a specific AP. A channel selection mechanism that considers channel conditions as well as AP load was proposed in [75]. 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 [76], 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 [77] 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 [70], a rather simple implementation is presented that requires a wired distribution system to operate. The work in [74] includes only limited testbed results, without providing any issues related to the implementation efforts. On the other hand, the work in [72] implements a simplified version of their algorithm, using precomputed range and interference sets in the experimental evaluation. The work in [73] 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.

4.1.2 *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 MadWiFi open source driver [23]. 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.

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

Table 4: 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

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.

4.2 INTERFERENCE MODEL

4.2.1 *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 [78] 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:

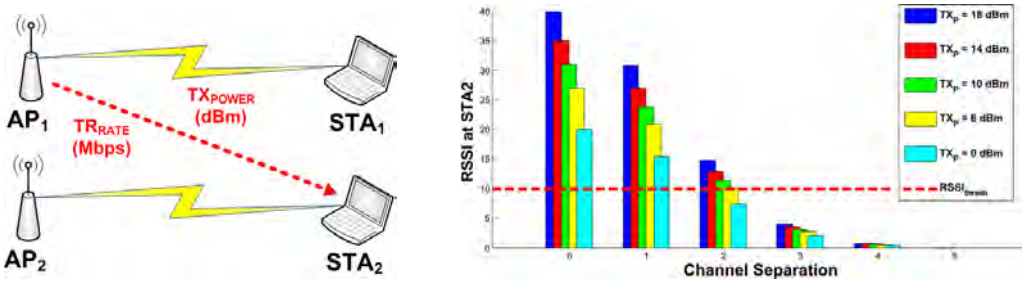
$$RSS_j(m) = RSS_j(n) * I(m,n) \tag{11}$$

Table I lists the measured values of I_{factor} for various channel separation values ($|m-n|$), as used in [78] and it is assumed to be zero for channel separation greater than 6.

4.2.2 *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 transmissions 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.



(a) Experimental Setup.

(b) RSSI - Channel Separation.

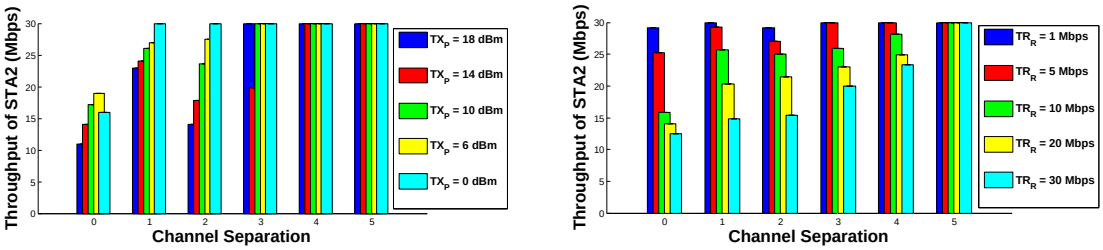
Figure 26: Motivating Experiment.

4.2.3 *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).

4.3 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. 26a, 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



(a) Throughput - Transmission Power of AP1.

(b) Throughput - Traffic Rate of AP1.

Figure 27: Throughput Performance of Motivating Experiment.

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 [78], as presented in Table I. Fig. 26b 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 26b, while Fig. 27a 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. 27b, 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.

4.4 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:

$$\sum_{j \in \mathcal{B}} RSS_j(n) * I(m, n), \tag{12}$$

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:

$$COT(m) = \sum_{k=1}^{F_m} \frac{L_k}{R_k}, \tag{13}$$

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:

$$\sum_{j \in \mathcal{B}} RSS_j(n) * I(m,n) * COT(m) \tag{14}$$

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:

$$\frac{1}{|\mathcal{A}|} \sum_{i \in \mathcal{A}} \sum_{j \in \mathcal{B}} RSS_{ij}(n) * I(m,n) * COT(m) \tag{15}$$

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.

4.5 PROTOCOL DESCRIPTION

IEEE has specified the 802.11h [79] 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.

4.6 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 adapter's 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. 28.

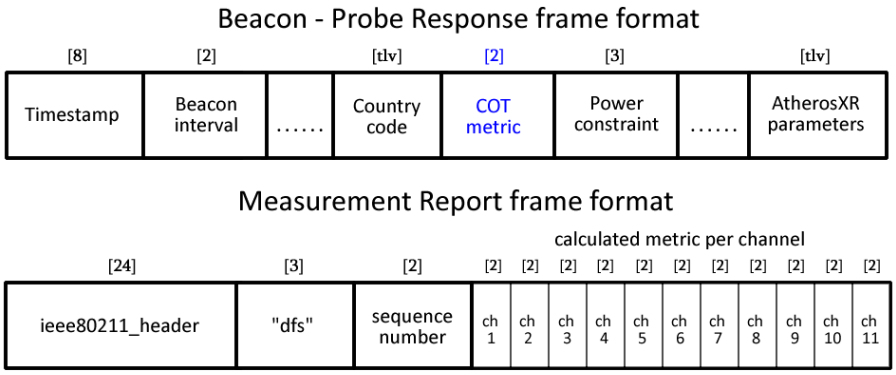


Figure 28: Frame format.

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.

4.7 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 [80].

4.7.1 *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.

4.7.2 *Measurement Methodology*

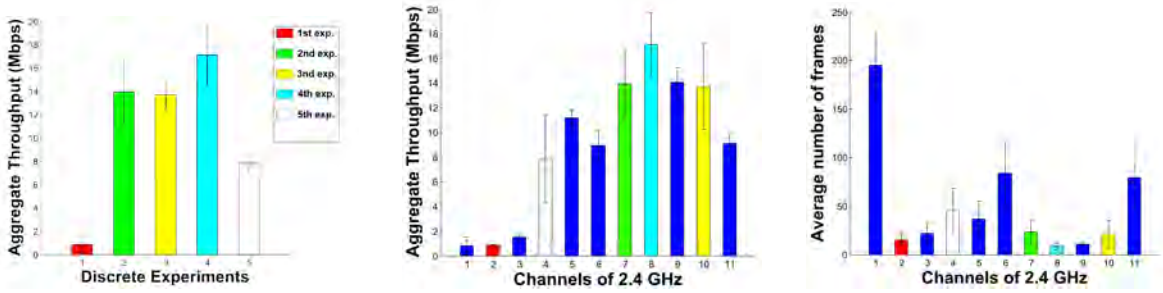
The throughput performance of the experiments is measured by using Iperf. A typical experimental setup for experiments considering only downlink transmissions, would be to run an Iperf client at the nodes, that act as APs, in order to generate traffic streams, having an Iperf server residing on each 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.

4.8 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.

4.8.1 *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.



(a) Throughput - Discrete Experiments.

(b) Throughput - Channels of 2.4 GHz.

(c) Monitored frames - Channels of 2.4 GHz.

Figure 29: First set of Experiments - Downlink Outdoor Experiments in the 2.4 GHz band.

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. 29a.

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. Moreover, we observe that the performance of the default algorithm is not efficient at all, which observation indicates that additional factors must be considered as well.

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.

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. 29a, measurements collected by the limited number of 5 associated STAs do not lead to notable throughput increase.

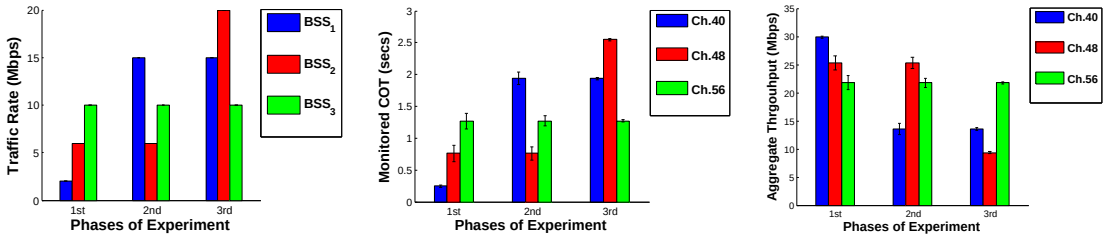
In the fourth experiment, we utilize the feature of our implementation that enables STAs belonging to neighboring cells to exchange *Measurement Report* frames. We activate a new part of the network that consists of 2 BSSs, each of them having 1 AP and

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. 29b 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. 29c, 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.

4.8.2 *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 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. 30a shows the traffic rate used by each STA in each discrete phase. Moreover, the aver-



(a) Traffic Rate in each BSS - Phases. (b) COT on each channel- Phases. (c) Throughput of BSS4 - Phases.

Figure 30: Second set of Experiments - Uplink Outdoor Experiments in the 5 GHz band.

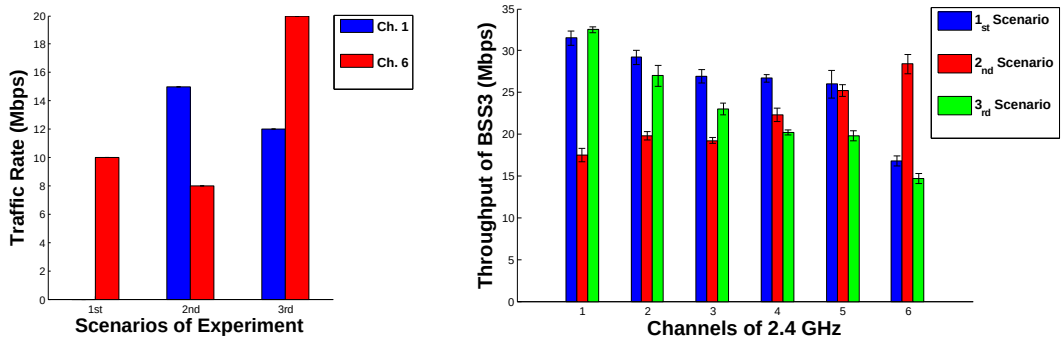
age COT values, as reported by the corresponding APs in their *Beacon* frames among the phases of the experiment, are portrayed in Fig. 30b.

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. 30c 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.

4.8.3 *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



(a) Traffic Rate of each BSS - Scenarios.

(b) Throughput of BSS 3 - Scenarios.

Figure 31: Third set of Experiments - Downlink Indoor Experiments in the 2.4 GHz band.

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. 31a 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. 31b. 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 across all the considered 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.

4.8.4 *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 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.

4.9 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 [81].

5

DETECTION AND MONITORING OF INTERFERENCE CONDITIONS IN WIRELESS TESTBEDS

5.1 INTRODUCTION

Over the past few years, wireless technology has evolved dramatically. The demand for wireless experiment infrastructure rises accordingly. It is typically time consuming and inefficient to build experiment setup each time for individual researchers. Therefore, many research institutes choose to use testbeds — a group of fixed and public accessible infrastructures for experimentations. Such a testbed not only shortens the experiment setup time, it also offers more resources, hence enables more sophisticated experiments.

In a wired-network testbed, the link configuration between network entities is part of the resources that can be reserved by users. Compared to the wired-network testbed, the wireless media is shared by all the facilities inside the wireless testbed, therefore also shared by different experiments from different users. As a result, a wireless network experiment performed on a large scale testbed has a possibility of getting interfered by other experiments. Such kind of interference between different users can not be solved by resource allocation like in the wired-network testbed. Apart from interference among users, nearby wireless devices, which are not part of the testbed, can also cause interference. Achieving stability is crucial for network experiments. However, the wireless environment is highly unpredictable such that conducting similar experiments and expecting identical outputs is rarely the case. As an example, a WIFI throughput experiment with fixed traffic pattern is repeated 7 times on the same channel. No interferer is expected during the measurement period. The results are shown in Figure 33. We can see that there is a fairly large variation among the results obtained in different rounds of the experiment.

The question is what is causing those variations? With such a simple experiment scenario, it is easy to conclude the variations are not from the experiment itself, but due to other external interference. For more complex experiments, such variations can be extremely confusing and eventually lead to wrong conclusions.

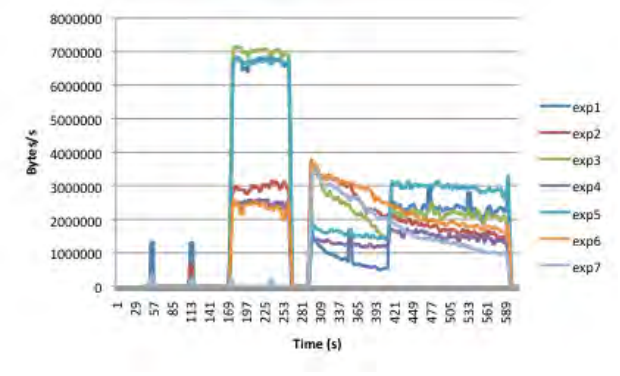


Figure 32: The maximum WI-FI bandwidth as a metric for channel stability over night

Figure 33: The w-ilab.t node

In short, we need a god view in the testbed to tell what is really going on in the air, which experiment is valid and which one is not. Hence, apart from normal experiment, it is necessary to have a monitoring system running in the background to detect undesired interference. More specifically, for a given experiment, the environment monitoring system should be running in three phases : before the experiment, during the experiment and also after the experiment.

- The monitoring before the experiment provides an overview of the channel condition. If any interference is detected during this phase, then perhaps the experiment should wait until the environment is clean, or simply consider to switch to another channel. The aim of this phase is to prevent invalid experiments from happening. A general energy detection is sufficient for this purpose.

- The monitoring system required during the experiment should be able to distinguish the interference from the ongoing experiment. In this case energy detection might be inadequate while feature detection is more desired.

- The post-experiment monitoring is similar to the pre-experiment monitoring phase, which requires only simple energy detection. The logic here is, if there is interference immediately detected after the experiment, then most probably it also happened during the experiment, hence we should give less credit on the validity of this experiment.

Since the pre-experiment and post-experiment monitoring only requires simple energy detection. The main challenge is to find the optimal technique to detect interference during the experiment.

Figure 34: The w-ilab.t Zwijnaarde testbed

This paper describes several techniques for monitoring the wireless environment during an ongoing experiment an OMF enabled testbed. The remaining part of the paper is organized as follows. Part 2 gives a high-level overview of the w-ilab.t testbed. Part 3 introduces the available monitoring facilities. Part 4 describes the monitoring techniques in depth combined with experiments. Part 5 presents an extra case of how distributed spectrum analysis can contribute to the wireless environment monitoring. Part 6 concludes this paper.

5.2 THE W-ILAB.T TESTBED

The w-iLab.t testbed is a generic and heterogeneous wireless testbed. It consists of two sub testbeds : the w-ilab.t office and w-ilab.t Zwijnaarde. The w-ilab.t office is deployed in a real office environment. The testbed Zwijnaarde is located at a utility room where no regular human activity should appear, in addition, the Zwijnaarde testbed has a pseudo shielded environment — most part of its wall and ceiling are covered with metal material , shown in Figure 34. This helps to reduce the amount of external interference. Unlike in w-ilab.t office, where WI-FI experiments are limited during the day time, the testbed Zwijnaarde provides a permanent testing environment. The majority of devices in both testbeds are embedded PCs equipped with Wi-Fi interfaces and sensor nodes. Since the Zwijnaarde testbed deployed more recently, all devices in this testbed are more powerful in terms of processing power, memory and storage. In this paper, our experiments are performed at the Zwijnaarde testbed, therefore from this point on we mainly focus on the introduction of the this testbed.

There are several different types of wireless devices deployed : Zigbee sensor nodes, Wi-Fi based nodes, sensing platforms, and the software-defined radio USRP N210 [82].

Figure 35: The Zwijnaarde testbed topology

The wireless nodes are also connected over a wired interface for management purposes. Each of the devices can be fully configured by the experimenters. When the different wireless devices are configured via the same control interface, they are said to be attached to one “node".

A common node in W-ilab.t Zwijnaarde is shown in Figure 32, which contains an embedded PC with two WI-FI interfaces, one Zigbee sensor node. Nodes are located at 60 spots throughout the utility room. The topology of the deployed nodes are shown in Figure 35. The USRP’s are attached to quadcore servers, there are 6 USRP’s in total, distributed over the testbed. The location of the USRP’s are marked with red dot in Figure 35.

The w-ilab.t Zwijnaarde has adopted OMF as its testbed control and management framework [26]. The architecture of OMF is based on three main software components: the Aggregate Manager (AM), the Experiment Controller (EC) and the Resource Controller (RC). The AM provides a set of services to the testbed (inventory, image loading, etc.). The EC, which is the user’s interface, receives and parses an experiment script describing configuration of resources and the actual experimental scenario. This script is written in a domain-specific language called OEDL (OMF Experiment Description Language). The instructions in the script are transferred to the RCs of the respective resources, which are responsible to perform the local configurations and application invocations.

OML (OMF Measurement Library), a companion framework for OMF, is responsible for handling measurements. It consists of two architectural components, the OML server and the OML client libraries. The client libraries are responsible for capturing measurements generated at the resources and, possibly after some manipulations, injecting them in streams headed towards the OML server. The OML server receives the data and stores them in organized databases, one per experiment.

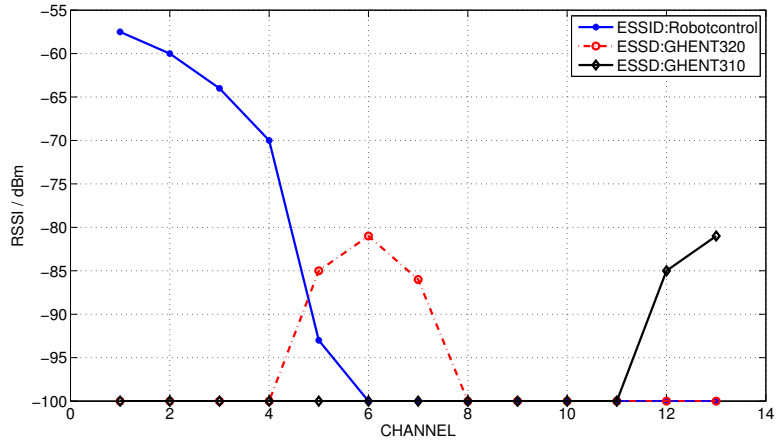


Figure 36: RSSI level as captured by Wi-Fi card in monitor mode

There are two main advantages of using OMF framework in the aspect of wireless environment monitoring system — its central control capability and data collection service. The central control capability is especially valuable when the system requires distributed spectrum sensing. As an experimenter, the monitoring tools are no more than regular experiment facilities, which can be controlled via the OEDL script. The data generated by monitoring system can be logged into the database just like regular experiment data. For an experienced OMF testbed user, the extra effort of using such a monitoring system is trivial.

5.3 AVAILABLE MONITORING TOOLS

5.3.1 *Use WI-FI Interface in Monitor Mode*

As mentioned before, a common node in the Zwijnaarde testbed has two WI-FI interfaces. Since most experiments do not utilize the second WI-FI interface, it is possible to configured the idle interface into monitor mode on the selected experiment channel. When configured into the monitor mode, the we can obtain information for all the packets that the WI-FI card is able to decode. Thanks to the Radiotap[83] header contained in the 802.11 frame, we are able to see the RSSI of each decoded packet and the physical layer bit rate at which the packet is received. Also the information from MAC layer protocol and above can be recorded.

The monitoring with WI-FI card is very technology dependent, but for interference on the same channel and same technology, it provides optimal detection

For the pre-experiment environment monitoring, the WI-FI card can be used to find out which access point is active on what channel, and how their power is distributed on the neighboring channels. As an example, one node in the Zwijnaarde testbed is configured to scan over the 13 WI-FI channels in 2.4 GHz ISM band. Figure 36 shows the results. We can tell that there are three access points active in the neighborhood, located on channel 1, 6, and 13 respectively. The beacons from access point robotcontrol on channel 1 has highest power, because it is located within the testbed. It is also shown that the beacons from channel 1 can be measured even up till channel 5. The other two access points have considerably weaker signal strength due to the fact that they are located out of the testbed. It is fair to say, when trying to select a channel for WI-FI experiment, it is safer to choose a channel above channel 5.

5.3.2 *USRP Based Spectrum Sensing Engine*

The Universal Software Radio Peripheral (USRP) developed by Ettus Research [82] is a low-cost SDR platform that utilizes a general purpose processor. USRP consists of two parts, a fixed mother board and a plug-in daughter board. The mother board mainly contains ADC/DAC, an FPGA mainly for digital down sampling with programmable decimation rate and an interface connected to host PC. The daughter board provides basic RF front-end functionality. In the testbed, all USRP's are equipped with XCVR2450 daughter board. The XCVR2450 is a daughter board that covers the 2.4 and 5 GHz ISM bands and has a configurable analog front-end filter with maximum bandwidth of 30 MHz.

We have implemented a customized component within the Iris platform [84] to use USRP for spectrum analysis. The collected spectrum information is power spectrum density (PSD), obtained via periodogram. By default, USRP's are configured to cover 25MHz, which is wide enough to cover one WI-FI channel. It is also possible to configure the USRP into a wide-band mode in order to get an overview of all the channels. In this mode, the front end of USRP is configured to do fast sweeping across the selected channels. However, the probability of interception on each channel decreases with the increase of covered bandwidth. More details about the implementation of USRP spectrum analysis is illustrated in [85]. A dedicated OMF wrapper is written for logging the PSD data into the database on AM.

The USRP sensing engine belongs to the energy detection category. It is technology independent, however the provided information is only on physical layer.

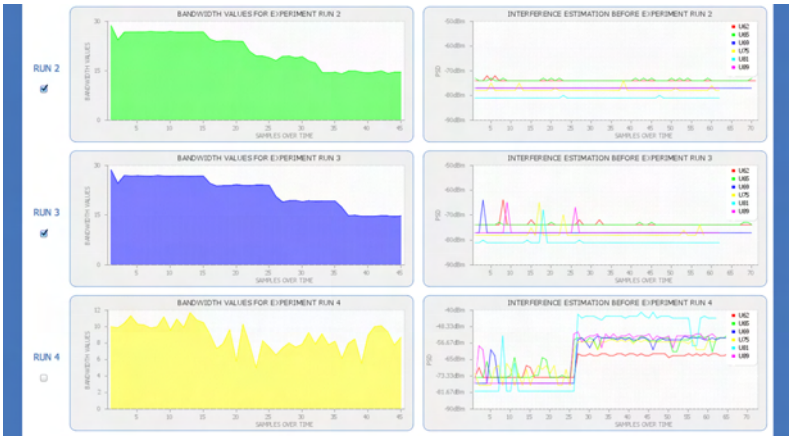


Figure 37: Visualization support

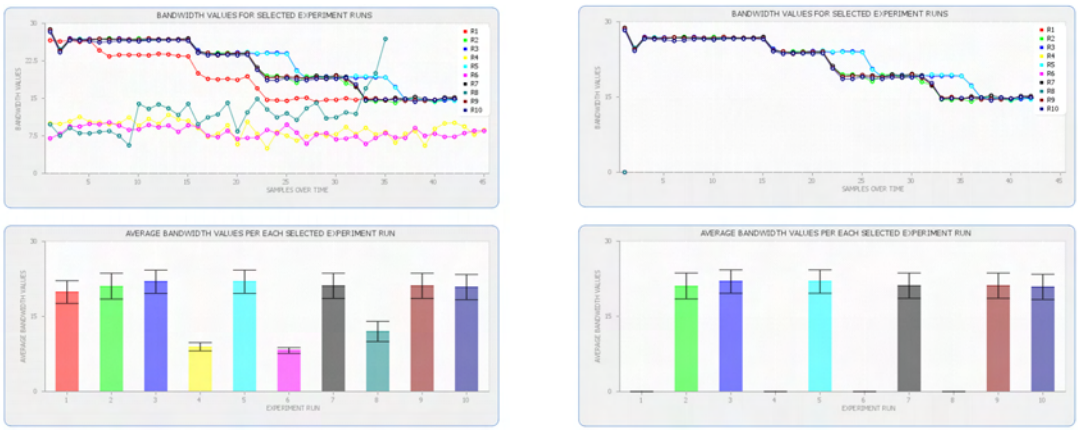
5.3.3 *The CONCRETE Framework*

The developed framework enables the experimenter to run several runs of the specified experiment and analyze the corresponding results. The main functionalities that are currently supported, are:

- scheduling the execution of several runs for the same experiment
- visualization of prevailing channel conditions before each run and moreover visualization of the performance achieved in each run
- visualization of performance over all executed rounds
- estimation of correlation among the different runs, as the benchmarking score that describes the stability of each run, accompanied by an automatic mechanism that selects the most stable runs, based on the calculated score
- comparison of performance obtained over all executed rounds in comparison with the performance achieved only in the subset of selected rounds.

Below, each different functionality is briefly detailed and is also accompanied by some representative screenshots.

As for the first step, the experimenter uses a custom Graphical User Interface (GUI) to provide all input required to describe the experiment to be executed. The framework proceeds with the execution of multiple experiment runs and also instruments the measurement gathering and storage procedures, through the use of the OMF (cOntrol and Management Framework) [26].



(a) Performance plot over all executed rounds (b) Performance plot over stable rounds

Figure 38: Representative screenshots

As for the second step, the framework accesses data stored during the execution and generates screenshots that illustrate both performance in each different round as well as interference conditions, before each discrete run. The interference monitoring is based on energy detection applied through distributed spectrum sensing, by utilizing USRP devices. Based on the use of different specified metrics (packet loss, ping delay, RSSI levels), we can get different plots for each specified experiment. In the left part of Fig. 37, we can see a representative plot of throughput measurements over 3 rounds, while on the right part, we can see a representation of the channel conditions over time, in terms of signal strength in dBm values.

In the next step, the tool constructs a joint performance representation plot over all runs, which also presents average and standard deviation values for each different run. Fig. 38a illustrates a representative screenshot.

Apart from providing simple statistics that result in an overall comparison over the various executions, an automated process for detecting whether the experiment is affected by external factors has also been implemented. The effect of external factors, such as interference, constitute the temporal correlation between each consecutive experimental run. Thus, the degree of correlation between multiple executions of the same experiment may vary according to the temporal variation of factors that impact the testing environment. Based on the estimated degree of temporal correlation among the multiple executions of the same experiment, the tool is able to provide a list with the runs that are considered stable. Fig. 38b illustrates a representative screenshot, where only performance of stable rounds is plotted. Based on the final set of selected rounds, a final average performance and standard deviation is calculated across the subset of selected rounds. Series of different experiments conducted in both w-ilab.t [86] and NITOS [80] testbeds have verified the great performance difference between cases where all runs are taken into consideration in comparison solely with stable runs.

5.4 EXPERIMENTAL EVALUATION

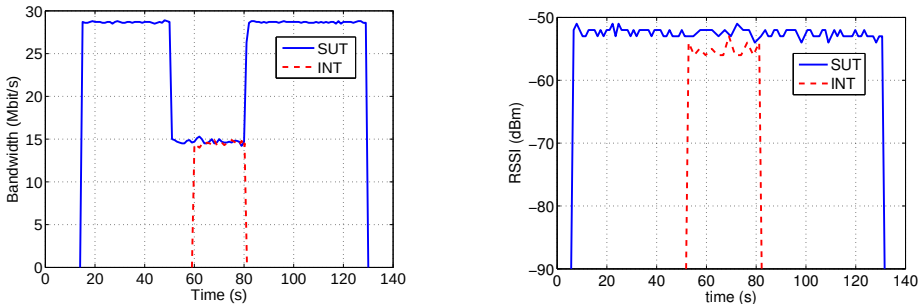
In this section, we evaluate several experimental scenarios that aim at presenting the abilities offered by the different interference detection tools and techniques. More specifically, these experiments have been designed to demonstrate some representative scenarios, in which each heterogeneous monitoring tool targets the detection of different types of interference.

In the experiments that follow, we consider a typical scenario of two IEEE 802.11 compliant nodes, operating in infrastructure mode on channel 11 (2452 MHz) and generating traffic on Uplink. We refer to these two nodes as the System Under Test (SUT) and also consider the measured UDP throughput performance of the SUT, as the overall performance metric. The first three experiments consider channel contention, overlapping channels and co-channel interference accordingly. The different types of interference are generated by a collocated pair of IEEE802.11 compliant nodes and under various settings, specifically designed for the needs of the different scenarios. In the fourth experiment, we use a narrow band signal that is generated by a USRP node and emulates a Zigbee or Bluetooth compatible device, in order to generate interference conditions among devices of heterogeneous technology. We refer to these two different types of nodes that generate Interference conditions, as the Interference (INT) generating nodes.

For all the aforementioned scenarios, we configure the second IEEE802.11 interface on the receiver of the SUT in monitor mode and continuously monitor the RSSI (Received Signal Strength Indicator) of all the packets that are successfully decoded. Moreover, in each scenario we configure a different number of USRP devices to perform spectrum sensing on the operating frequency of the SUT and also on other adjacent channels, in order to record PSD values for 10 sweeps per second, under each specific setting.

5.4.1 *Channel Contention Detection*

In this first experiment, we place the INT generating pair of nodes close to the SUT and configure it to operate on the same channel as the SUT, channel 11. These two pairs of nodes are indicated with green circles in Figure 35. The nodes in the top green circle belong to the SUT, while the nodes within the bottom green circle represent the INT group. We use the Iperf to generate traffic on the application layer and also set the bandwidth level of both the SUT and INT flow to 30 Mbit/s. The Iperf of the SUT group is active throughout the duration of the experiment, while the Iperf



(a) Throughput performance of SUT (b) RSSI trace of WI-FI card

Figure 39: Channel Contention Detection Experiment

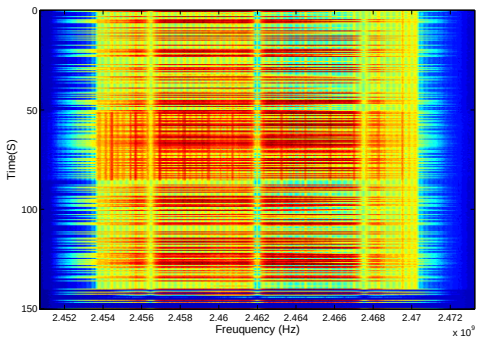
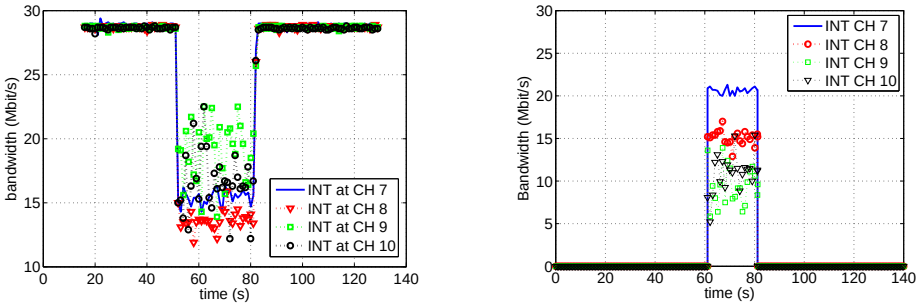


Figure 40: Spectrogram of USRP4 sensing channel 11

application of the INT group is activated just for 25 seconds, in order to show how the performance of the SUT is affected. The throughput performance of both the SUT and the INT group is illustrated in Fig. 39a. Upon the activation of the INT group, we notice that the throughput performance of the SUT drops to the half (15 Mbit/s) and equals the throughput of the INT, which clearly shows that the available channel capacity is equally distributed through the CSMA protocol, between the two contending pairs of nodes.

Having examined the bandwidth performance of the SUT, the next step is to check the performance of the monitoring tools. Through the processing of the RSSI records, obtained by interface in monitor mode, and moreover through source MAC address filtering, we illustrate the RSSI results for each different group in Fig. 39b. Hence, for this scenario, a simple monitor interface is able to clearly identify which IEEE802.11 compatible device and when exactly it is contending for channel use. This feature can only be made possible through the identification of the source of the interference, based on MAC address filtering.

However, the USRP devices, which are not aware of MAC layer information, are not able to distinguish between different Wi-Fi sources that transmit on the same channel. We easily notice in Fig. 40 that the PSD measurements gathered from USRP4, do not provide any valuable input that can aid in the discrimination of the different traffic



(a) SUT (b) INT

Figure 41: Throughput performance per node pair

sources. The only valid observation is that the channel under consideration is occupied during the entire experiment and more specifically that the density of the spectrogram becomes slightly higher, while the INT group is active. However, such observations are not sufficient to detect interference activity and therefore, energy detection is not suitable for detecting channel contention conditions.

5.4.2 *Overlapping Channels Interference Detection*

The second experiment is designed to evaluate the effect of interference, which is generated between IEEE802.11 compliant devices that operate on overlapping channels. For this purpose, we use the same network setup that was used in the previous experiment. However, instead of configuring the INT group to operate on the same channel as the SUT, we set it to transmit on adjacent channels and more specifically vary the channel of operation from channel 7 to 10.

In Figure 41a and Figure 41b, we illustrate the throughput performance yielded by the two pairs among the various channels used by the INT pair. We notice that the activation of the INT link results in significant reduction of the SUT performance, in all the cases under consideration. In addition, we notice that while the INT link is active, the bandwidth performance on channel 9 and 10 is slightly higher than it is on channel 7 and 8. This comes from the fact that when the amount of channel overlapping falls below a certain threshold, then the carrier sense mechanism fails to detect other ongoing transmissions and thus results in packet collisions. However, among channels that are closely spaced, such as Ch. 9, Ch. 10 and Ch. 11, we notice that the maximum available channel capacity is still obtained, due to the successful contention for channel use.

Another important remark is that there exist specific settings, under which the INT link yields higher performance, in comparison with the SUT link, as in the cases of Ch. 8 and Ch. 10. This phenomenon is related to the “Capture Effect" [], which occurs

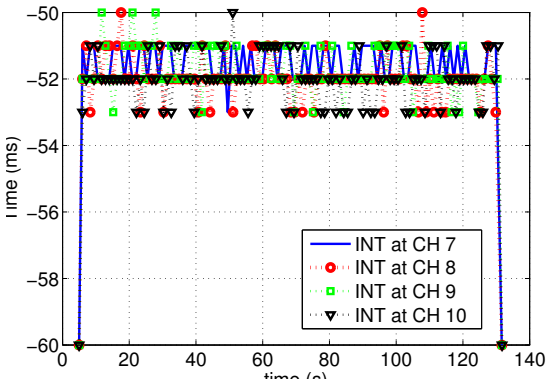


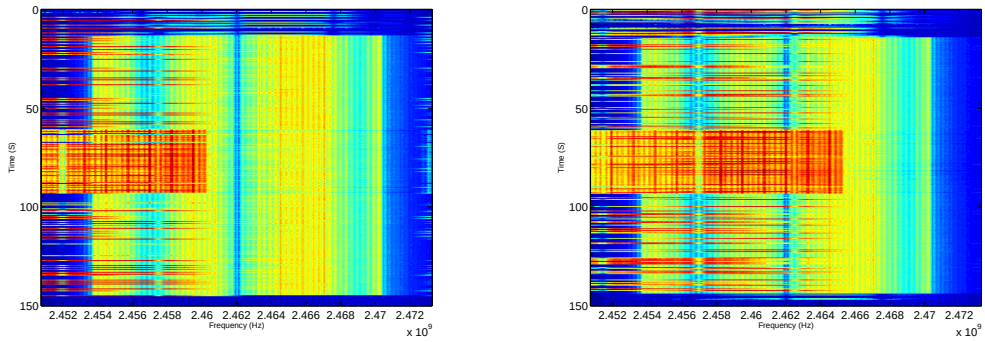
Figure 42: RSSI Trace of monitor mode

in cases where frames received with high RSSI suffer only few losses, as opposed to frames that are simultaneously received, but with relatively low RSSI values. According to our experiments, certain topology and channel configurations lead the capture effect to either favor the SUT or the INT link. These observations yield interesting insights regarding the impact of the “Capture Effect" on interference and motivate further investigation.

In Figure42, we plot the RSSI records obtained with the monitor interface. Unlike the previous experiment, we are not able to decode any packets transmitted by the INT link and as a result, all the recorded RSSI measurements in this case correspond to the SUT transmitter node. Hence, the Wi-Fi compatible monitor fails to detect any interfering activity for this scenario. Considering the measurements that were plotted in Figure 5, we notice that Beacon frames were decoded even between channels that are not adjacent. This can be explained based on the fact that Beacon frames are Management frames and thus are transmitted at the basic rate of 1 Mbit/s, while the data from application layer is transmitted at much higher rates, typically 24 Mbit/s or above. Hence it is much easier to successfully decode the Beacon frames, than regular data frames.

Based on the above observations, we clearly conclude that standard compliant devices that operate in monitor mode, are not able to provide valid measurements, in order to detect Co-Channel interference successfully. Fortunately, we can overcome such situations, by using another “eye" in the air — the USRP sensing engine. Figure 43a and Figure 43b represent the recorded spectrogram when interference is present on Ch. 9 and Ch. 10 respectively. Based on the spectrogram, it is clear that activity in adjacent channels is the reason for the SUT throughput reduction.

Furthermore, through the USRP platform, we were able to detect Beacon frames transmitted by the INT group’s access point, before the activation of the INT transmitter node. The access point of the INT link is closer to the USRP sensing engine and as a result, the Beacon frames of the INT link are detected stronger than the beacons from



(a) Channel 9 (b) Channel 10

Figure 43: Spectrogram per Channel

SUT. However, despite the strong Beacon signals, the bandwidth of SUT appears to be regular when no data traffic is present.

5.4.3 *Co-channel Interference Detection*

This experimental scenario focuses on how distributed sensing can contribute to the detection of co-channel interference. As previously mentioned, co-channel interference may occur when the two devices that interfere are located far away and operate on the same channel. As a results, they fail to detect each other's ongoing transmissions and therefore start transmitting simultaneously, resulting in packet collisions. A widely used technique that targets detection of co-channel interference is distributed sensing. For the purposes of this experiment, we setup a distributed spectrum sensing system, which is based on the 6 USRPs that are currently distributed over half of the testbed and also on their orchestration through the OMF framework.

Similar to the experimental setup used in the previous scenario, the experiment here also involves two pairs of IEEE802.11 compliant nodes, however instead of choosing two groups next to each other, the groups are now located at different sides of the testbed. More specifically, we designed a specific topology that is able to reproduce "Hidden-Terminal" effect. When facing the testbed's topology (Figure 35), the selected nodes for this experiment are marked with red dots, labeled from 1 to 6.

We configured the SUT group on the left side of the testbed to generate continuous traffic and set the transmit power equal to the default value of 20 dBm. The INT group on the right side of the testbed follows an ON-OFF traffic pattern, so that each time the client starts a data stream for 15 seconds, OMF will wait for another 15 seconds before the client is turned on again. We decided to follow this specific ON-OFF traffic pattern

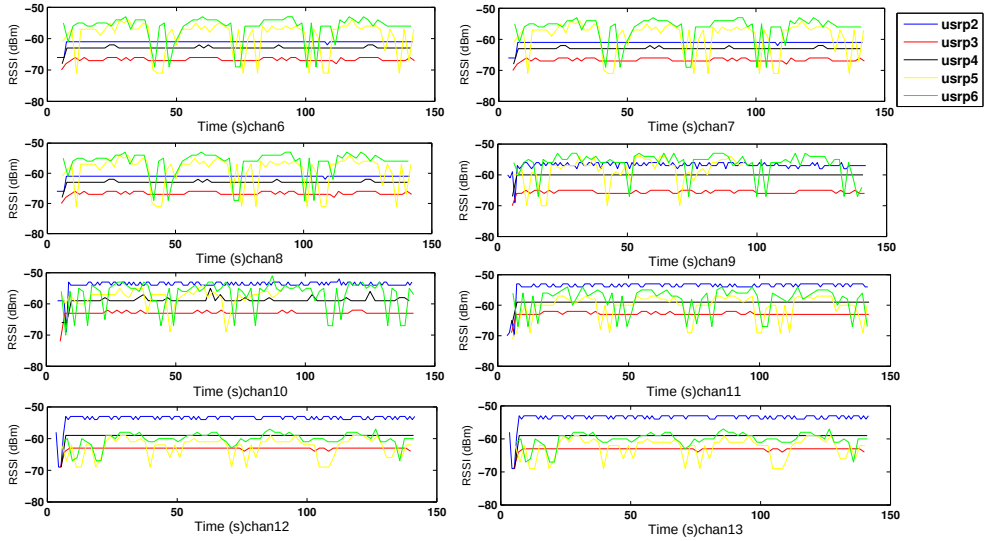


Figure 44: PSD vs Time plots for 8 channels

for the transmissions of the INT group, so that we could discriminate between signals that are generated by the SUT and the INT groups, through simple RSSI measurements.

Ideally the duty cycle time should be 50%. In reality the duty cycle is also influenced by the control lag introduced within OMF. The second group's transmit power is configured to 5 dBm. The left group is operating on channel 11 while the right group is operating on channel 9. Thus, the experiment involves diversity in both space and frequency domain.

Unfortunately USRP 1 was not available during the conduction of this experiment, we hence only configured the rest 5 USRPs to collect spectrum data. Each USRP sensing engine produces a max-hold PSD value every two seconds over the selected channel of the ISM band. The experiment is performed 8 times, where among different executions, we vary the channel that is being monitored by the USRPs from Ch. 6 to Ch. 13. Moreover, during each discrete experiment, we configure the 5 USRP sensing engines to monitor the same channel.

The collected PSD value is plotted against time for all the monitored channels in Figure 44. At the first glance, USRP 5 and 6 are able to follow the ON-OFF traffic pattern produced by the group on the right side, while the rest of USRPs are dominated by the left group's transmissions, since their PSD measurements appear to be more stable. During observation of the RSSI measurements across channels, we notice that while channel 9 is being monitored, USRP 5 and 6 clearly detect the on-off pattern that is followed by the INT link. However, the detection of this characteristic traffic

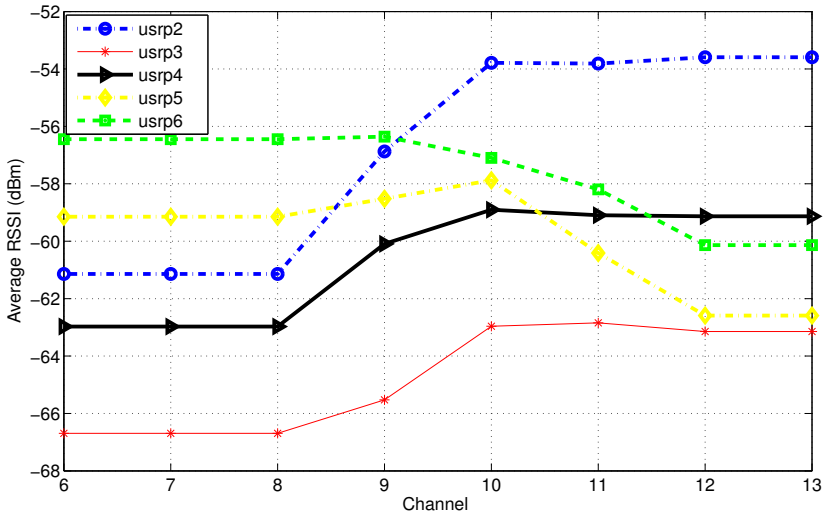


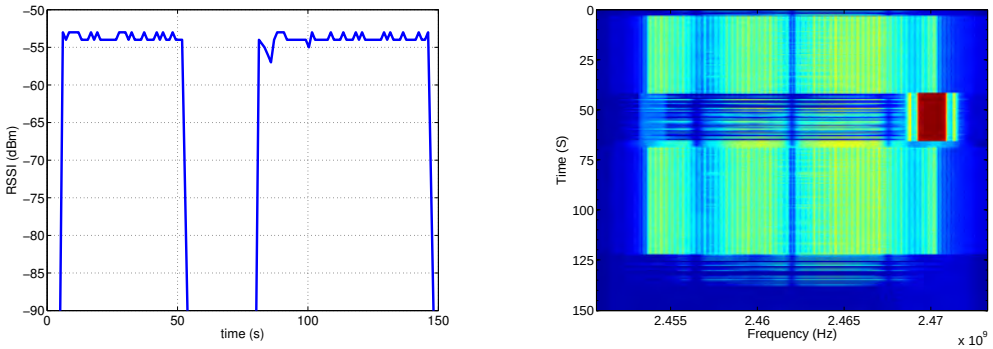
Figure 45: The average PSD vs Channel

USRP ID	2	3	4	5	6
2	1.0	0.5363	0.5217	0.0167	-0.1665
3	0.5363	1.0	0.7941	0.1789	0.0376
4	0.5217	0.7941	1.0	0.0484	0.0135
5	0.0167	0.1789	0.0484	1.0	0.0967
6	-0.1665	0.0376	0.0135	0.0967	1.0

Table 5: Correlation of measurements collected by different USRP devices

feature cannot be guaranteed in the cases that the monitored channel drifts far from the operating channel of the SUT group.

On the other hand, for the rest of the USRP nodes, we notice that the measured signal strength increases while the monitored channel's index gets closer to the operating channel of the left group. This trend is clearly reflected in Figure 45, where the average PSD value is plotted against channel index values. In order to further discover the relationship among measurements obtained by different USRP's we use the CONCRETE framework and list the collected results in Table 5. We clearly observe that measurements collected by USRPs 2, 3 and 4 are highly correlated, as the three devices are closely spaced and thus experience similar experimental conditions.



(a) RSSI measurements through monitor node (b) Spectrogram through USRP device

Figure 46: Heterogeneous Technologies Interference Detection Experiment

5.4.4 *Heterogeneous Technologies Interference Detection*

In this last experiment, we do not use IEE802.11 devices for the generation of interference conditions, nut we make use of an interference signal of heterogeneous technology and more specifically, we use a narrow band signal that is generated by a dedicated for this purpose USRP device. The interfering signal is 5 MHz wide, centered at 2.47 Ghz. This jamming signal is a general representation for interference generated by none IEE802.11 compliant devices, such as zigbee or bluetooth ones.

Through throughput performance measurements, we observe that due to the activation of the narrow-band jammer, the performance of the SUT almost drops to zero, while the jitter increases linearly with time. The RSSI trace recorded by a monitor node, illustrated in Fig. 46a, confirms this observation. Since no packets are decoded, no RSSI is recorded during this period.

However, through the spectrogram that is obtained via the USRP sensing engine, in Fig. 46b, we clearly see that the narrow band jammer resulted in the pause of the SUT transmissions. Based on this fact, we notice that the IEEE802.11 transmitter detects that there exist ongoing transmissions through the Carrier Sense mechanism, and thus defer from transmitting for the whole duration of the jamming signal. Although in this case, the monitor node is not able to provide any extra information regarding the jammer, it is at lest able to detect the abnormal physical layer behavior of the SUT itself. However, we also noticed that for less strong jamming signals, the detection based on monitor nodes becomes difficult and even unsuccessful.

5.5 CONCLUSION AND FUTURE WORK

The complexity of large scale wireless experiment facilities or testbeds is one of the hurdles an experimenter will face during solution buildup. The methodologies for conducting an experiment vary widely from testbed to testbed In this paper, we prove that wireless experiments are very susceptible to unpredictable interference, a monitoring system is hence necessary to validate the experiments via three phase . Two types of measurement tools are examined: the standard WI-FI card, and the custom-designed sensing engine based on the software-defined radio platform (USRP).

We considered several common interference scenarios for WI-FI experiments inside the wireless testbed, including the interference caused by WI-FI device on the same channel, the interference caused by WI-FI device from overlapping channel and finally interference caused by none WI-FI technology. For each scenario, the performance of each measurement tool are examined. Finally a set of distributed sensing experiments are performed. Through this set of experiment, we are able to see the that the results from some sensing engines are highly correlated and some are not. The degree of correlation is mainly based on the spatial location.

6

ONLINE EVALUATION OF SENSING CHARACTERISTICS FOR SPECTRUM SENSING PLATFORMS

6.1 INTRODUCTION

Cognitive Radio Networking is a rapidly evolving research thrust area in wireless communications nowadays, aspiring to create a major paradigm shift in the wireless landscape through Dynamic Spectrum Access and Management. The dynamic nature of the wireless medium, in accordance with the distributed frequency adaptation mechanisms, as applied in unlicensed bands, necessitate the development of spectrum-agile radios. In this context, spectrum occupancy monitoring is considered as the most important component towards enabling spectrum aware radio operation.

Theoretical developments [87, 88] in the area have proposed various methods for identifying the presence of signal transmissions and evaluating spectrum occupancy. In addition, several realistic experimentation platforms [89, 90, 91, 92] have emerged to enable the practical implementation of spectrum sensing functionality in real systems. Sound experimental validation of proposed spectrum sensing solutions requires experimentation under real world network scale and settings. To this aim, experimental cognitive radio testbeds [93, 94, 95, 80, 86, 96] have been deployed, while tools for orchestration of complex scenarios [26] and collection [51] of experimental data have also been developed. In order to extend cognitive testbeds with more advanced experimentation capabilities, existing tools have to be advanced and new frameworks have to be developed. In this work, we develop an evaluation framework that assesses the performance of cognitive platforms, in terms of two rather important metrics, namely sensing delay and energy efficiency. The former is directly related with the resulting performance of the cognitive solution, while the latter is considered as a crucial issue in all types of wireless communications, both due to restricted battery autonomy of mobile devices, as well as for moving towards "greener" solutions in telecommunications. Evaluation in terms of the aforementioned metrics is automated through a monitoring procedure that has been directly integrated in the experimentation tools of CREW [97] testbeds.

The remainder of this paper is organized as follows. In the following section we present the state-of-the-art spectrum sensing hardware and software platforms, referring also to well-established CR testbed deployments and the frameworks that are used to facilitate experimentation. Section 3 details the characteristics of the considered spectrum sensing devices, while also describing the experimental scenarios, under which performance evaluation of the various platforms will be based on. In Section 4, we present the various parts that constitute the proposed evaluation framework, while in Section 5 we present extensive experiments that compare the performance of 4 sensing solutions in terms of the considered metrics. Finally, in Section 6 we assess the performance of an external use case that implements real-time spectrum sensing on SDRs and point out the conclusions reached through this work in Section 7.

6.2 RELATED WORK

In an effort to support realistic and large-scale experimentation with spectrum sensing platforms, several CR testbeds have recently been deployed. In the US, existing experimental CR deployments include the Virginia Tech CR Network (VT-CORNET) [93], the ORBIT [94] and Emulab [95] testbeds. In addition, NITOS [80], w-ilab.t [86] and Cortex-lab [96] are the main wireless experimentation testbeds in Europe that are equipped with heterogeneous Software-Defined Radio (SDR) platforms. Among the various developed SDR solutions, USRP [89] is one of the most popular commercial hardware, while more sophisticated solutions, such as the WARP [90] and SORA [91] have also been proposed. On top of the underlying hardware components, software radio architectures have been developed to specify the interconnection and facilitate configurability of components that implement reconfigurable radios, such as GNU Radio [98], the Software Communication Architecture [99] and IRIS [100].

As most experimental testbeds are developed separately from each other, an important issue that experimenters traditionally had to cope with is the scarcity of a common management system, as well as a common experiment description language. However, this issue was overcome with the introduction of OMF, which provides tools for the management and execution of experiments on testbed infrastructures. Nowadays, OMF has been deployed and used on multiple testbeds supporting many different types of technologies. Among the aforementioned CR deployments, ORBIT, NITOS and w-ilab.t testbeds have adopted OMF as their control and management framework. In our previous works, we developed the NITOS EMF [101] and CONCRETE [102] frameworks that extend the core OMF functionalities, by enabling the experimenter to experimentally evaluate the energy efficiency of proposed protocols and to validate the stability of

experimental conditions accordingly. An important direction that we are also currently investigating is the direct integration of the IRIS software radio architecture with the OMF framework.

6.3 EVALUATION OF CREW TESTBED SENSING DEVICES

In this work, we exploit the advanced spectrum sensing platforms that are provided by the CREW federated testbed platform, which facilitates experimentally-driven research on advanced spectrum sensing, cognitive radio and cognitive networking strategies. Among the 5 individual wireless testbeds that constitute CREW, we decided to use the w-iLab.t indoor testbed, as it offers all the required hardware and software components for the development of the proposed framework. More specifically, w-iLab.t features several sensing devices that span from commercial sensor and Wi-Fi nodes, to SDR platforms and device prototypes. Another basic characteristic that motivated the usage of the w-ilab.t testbed, was the adoption of OMF as its testbed control and management framework, which fact enabled us to build the proposed framework as a plug-in compatible with the well-adopted OMF. In the rest of this section, we describe in detail the capabilities of the considered w-ilab.t sensing devices, while we also describe the reference scenarios, under which performance evaluation of the various platforms would be based on.

6.3.1 *Characteristics of Sensing Devices*

In our experiments, we consider the well-established commercial SDR USRP N210 Networked Series [103], which is a commercial SDR platform that utilizes the general purpose processor of a connected host machine for measurement processing and due to its applicability has gained widespread usage. In addition, we consider the embedded version USRP E110 [104], which allows standalone operation without requiring the use of a host machine, thus allowing us to evaluate how its limited internal processing capabilities impact sensing performance. The next device under consideration is the prototype imec Sensing Engine (SE) [92], which is a high-end prototype reconfigurable radio that is designed to process samples real-time. Finally, we also decided to experiment with the Wi-Fi compliant Atheros AR9380 [105] chipset, as this device is able to operate as a spectrum scanner as well. By including the AR9380 device in our experiments, we can investigate how well the hardware assisted FFT processing capabilities of this chipset perform, contrary to the rest research oriented sensing devices. More details about the capabilities of the devices under investigation follow below.

6.3.1.1 *USRP N210*

The USRP N210 consists of two parts, a fixed motherboard and a plug-in daughterboard. The motherboard contains ADC/DAC, an FPGA for digital down conversion with programmable decimation rate and a Gigabit Ethernet (GbE) interface for communication with the host machine. The USRP Hardware Driver (UHD) is used to control the USRP hardware and also to transmit and receive data over the GbE interface. We attach the USRP N210 with the XCVR 2450 daughterboard, which provides basic RF front-end functionality in the 2.4 GHz and 5.9 GHz bands. The channel switching delay for the XCVR 2450 daughterboard was measured and found to be 50 ms. Although the ADC is able to sample at the 100 MHz rate, the GbE interface limits the maximum streaming bandwidth to 25 MSps. In order to provide real-time sensing capability, the host machine should be able to achieve sufficient amount of parallel processing and thus this feature is host machine dependent.

6.3.1.2 *USRP E110*

The USRP E110 [3] series combines a flexible RF frontend, FPGA and an OMAP 3, which includes an ARM Cortex-A8 and a C64 DSP. The extra features differentiate the E110 from the N210, as the former does not require the use of a host pc for measurement processing and thus allows standalone operation for embedded applications, or applications that do not require the full processing power of a commodity CPU. The USRP E110 is also programmed through the USRP Hardware Driver (UHD) and attached with the XCVR 2450 daughterboard is able to operate in the 2.4 GHz and 5.9 GHz bands. The channel switching delay for this setup also equals 50 ms. The attached ADC is able to sample at the 64 MHz rate, but the FPGA interface provides for a maximum total bandwidth of 10 MSps. However, this does not guarantee that the embedded processor will be able to process samples at this rate. In the case that the collected samples are processed on an external host, the USB 2.0 interface limits the host bandwidth to 8 MSps, which results in the maximum streaming bandwidth of 5 MSps. In this case, real-time sensing can only be achieved if the host is able to perform efficient parallel processing.

6.3.1.3 *imec SE*

The prototype imec SE [4] is a reconfigurable radio and consists of two core components: an analogue RF front-end SCALDIO (SCAlable raDIO) and a DIgital front-end

for Sensing (DIFFS). Both these ICs are low-power and flexible and targeted towards implementing a cognitive radio as a mobile device. DIFFS features an FFT accelerator core, enabling the engine to perform spectral analysis. The receiver RF operating frequency is programmable from 0.1 to 6 GHz and the channel bandwidth is programmable between 1 and 40 MHz. The architecture of the Scaldio chip allows us to already handle the reprogramming for the next frequency band during the current capture. Hence the reconfiguration time is reduced to zero and only the settling time of 50 μs needs to be considered. Both the DIFFS processor and the ADC are running at 40 MHz. However, in our configuration the ADC is down-sampled to 20 MHz, resulting in the available channel bandwidth of 20 MHz. Hardware-wise the imec SE was designed to be able to process samples real-time. As the measurement collection and processing procedures can run in parallel, we just have to make sure that processing finishes before new data is available, in order to provide for real-time sensing. The imec SE firmware that is used in our experiments is able to perform all required calculations, before the next vector of samples that needs to be processed is available and thus is capable of real-time sensing.

6.3.1.4 *Atheros AR9380*

The Atheros AR9380 is a single-chip, dual-band (2.4/5 GHz), 802.11n compatible chipset that supports up to 3x3 MIMO transmissions, offering both high throughput performance, along with low power consumption in every operational state. The special characteristic of AR9380 is its ability to support spectrum sensing capabilities. The Ath9k [106] driver enables the user to configure the card, in order to sense the spectrum over a specified set of channels and derive the Power Spectral Density with an FFT resolution of 56 bins. The 56 bins correspond to the 52 usable plus the 4 pilot sub-carriers that are supported by the 802.11n protocol in the case that the 20 MHz bandwidth configuration is applied. The FFT operation is executed on the chipset and scanning results that correspond to 4μs long snapshots of the spectrum are reported per frequency at the fixed period of 55 ms. The channel switching delay was measured at 1 ms for in-band and 2 ms for out-band transitions. The channel switching overhead of the imec SE and AR9380 is obtained through the device specifications that are provided by imec and Atheros accordingly, while such values are not officially reported for the USRP devices and are derived through our own measurements. We experimentally verified that maximum configurable sampling rate equals 100 KSps. The driver provides an interface to configure both the actual sampling rate through adaptation of the decimation level at 255 levels, as well as the number of result sets that will be reported within the 55 ms interval. Although the AR9380 is able to sample at a sufficient sampling rate, this

Sensing device	Host requirement	Channel bandwidth (MSps)	Channel switching delay	Real-time sensing
USRP N210	Yes	25	50 ms	Host-dependent
USRP E110	No	5	50 ms	Host-dependent
imec SE	No	20	50 µs	Yes
AR9380	Yes	20	1-2 ms	No

Table 6: Hardware characteristics of sensing devices

device is not able to provide for real-time sensing, as continuous sampling can only last up to 55 ms, upon which interval the device goes back to regular Wi-Fi operation.

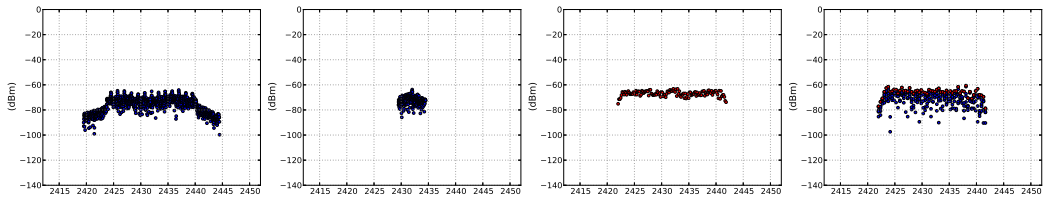
6.3.2 *Experimental Scenarios*

The various hardware platforms are used in parallel to detect a signal generated, under various settings, by a pair of w-ilab.t testbed nodes. The test signal consists of a bunch of 802.11 frames, as this is the most common signal transmitted in the 2.4 GHz and 5GHz ISM bands that can be sensed by all the different spectrum sensing platforms. As the latest widely adopted version of the IEEE 802.11 standard is the 802.11n version, the test signal consists of 802.11n transmissions that occupy the medium for significantly lower duration compared to legacy 802.11 transmissions, due to the increased PHY-layer transmission rates that 802.11n offers. Considering transmissions of typical MPDU frames of 1534 bytes length, frame transmissions duration approximates 227 µs and 63 µs at the highest PHY-layer rates of 54 Mbps and 195 Mbps that are supported by the 802.11 and 802.11n protocols at the bandwidth configuration of 20 MHz.

Having properly investigated the operation and capabilities of each different device, as listed in Table 6, we concluded that the only hardware dependent specifications that could restrict the design of the experimental scenarios would be related with the imec SE and the Atheros 9380 devices. The USRP devices are fully configured through software, thus generating no restrictions. Considering the real-time sensing capability offered by default by the imec SE, we decided to build a proper scenario which would allow for a fair comparison of the other devices with the high-end imec SE. Based on the device specifications that were analyzed in the preceding section, we know that the only hardware dependent restrictions are the 20 MHz sampling rate of the imec SE and its 128 bin size of FFT calculations. When the imec SE is sampling at this rate, it requires 6.4 µs, in order to gather 128 samples that are further fed to the on-chip FFT accelerator core for FFT processing that finally provides the PSD results. As the interval of 6.4 µs is extremely short, compared to the shortest duration event of 63 µs,

Sensing device	FFT resolution	Result sets	Collected samples
USRP N210	1024	1	1600
USRP E110	256	1	320
imec SE	128	10	1280
AR9380	56	6	336

Table 7: Sensing characteristics of sensing devices in the considered scenarios



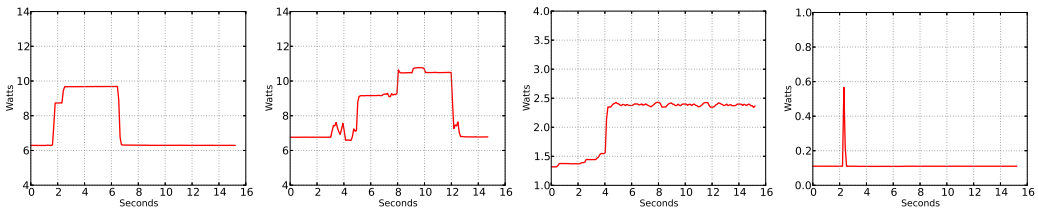
(a) USRP N210 (b) USRP E110 (c) imec SE (d) Atheros AR9380

Figure 47: Power Spectral Density Evaluation - Scenario 1

we further decided to combine several FFT measurements, before arriving at a final result. We use Max-Hold filtering over 10 consecutive FFT operations, which results in the total sensing time of 64 μs, within which interval we are able to decide about the presence of 802.11n transmissions.

Considering the fact that the aforementioned sensing devices feature varying sensing capabilities, we will test them under the reference scenario, to comparatively evaluate their performance in terms of sensing delay and energy efficiency. In order to provide for a proper evaluation and comparison setup, we decided to configure all the devices to sense for the same duration of 64 μs, which we denote by Measurement Collection Duration (MCD). Having specified a common MCD, the different devices will sense the medium for the same duration and collect a different number of samples, as specified by their sampling rate configurations. In the next phase, a common measurement-processing program will process the different number of collected samples and report the time required for processing.

Moreover, we also consider a second scenario, where the targeted band is 80 MHz wide, so that all devices will have to perform channel switching to provide for proper monitoring of the whole band. In this second scenario, the total sensing delay of all devices will further vary due to the impact of the channel switching overhead. Various characteristics as configured for each device in the considered experimental scenarios are listed in Table 7.



(a) USRP N210 (b) USRP E110 (c) imec SE (d) Atheros AR9380

Figure 48: Power Consumption Evaluation - Scenario 1

6.4 EVALUATION FRAMEWORK

In order to provide for a fair performance evaluation, between the 4 considered devices, we configure them under their operational limits to sense the medium and assess the Power Spectral Density (PSD) on the 2.4 GHz band. In parallel with the spectrum sensing operation, we evaluate the power consumption and the distribution of sensing delay per device. In the rest of this section, we describe in detail the 3 steps that constitute our evaluation framework.

6.4.1 *Power Spectral Density*

Spectrum sensing efficiency evaluation can be based on the quality of PSD characterisation. The PSD characterises the distribution of energy variations and more specifically quantifies the amount of energy per certain frequency interval (dBm/Hz). PSD computations can be obtained directly through FFT transformation of power results that have been collected over time. Frequency resolution is directly dependent on the sampling frequency and the FFT size. More specifically, for a fixed sampling interval, the frequency resolution can be improved by increasing the FFT size. However, the maximum configurable FFT-size depends on the maximum sampling rate of each device and is even hardware dependent for devices that perform FFT calculations in hardware. To conclude, an inherent tradeoff exists between the configuration of PSD parameters and the obtained resolution, which we aim to evaluate jointly with the proposed metrics of Sensing Delay and Power Consumption.

6.4.2 *Power Consumption*

In order to accurately measure the instantaneous power consumption, we follow the power measurement procedure and the hardware developed in our work [101]. Based on this approach, we place a high-precision, low impedance current-shunt resistor (R)

of a known resistance value, in series with the power source and the power supply pin of the device to be measured. More details can be found in Section 8.3. By consistently measuring the voltage ($V_R(t)$) across the current-shunt resistor through proper voltage metering equipment, we are able to extract the instantaneous current draw of the device, based on Ohm's law. The instantaneous power consumption can be calculated as the product of the input voltage V_{IN} and the measured current draw:

$$P(t) = V_{IN} \frac{V_R(t)}{R} \tag{16}$$

Estimation of the total energy consumption during a specific experiment, necessitates the accurate sampling of the instantaneous power consumption during the total experiment duration. Total energy consumption can be calculated as the integral of the power consumption over the specified duration ($Dt = t_1 - t_0$), as follows:

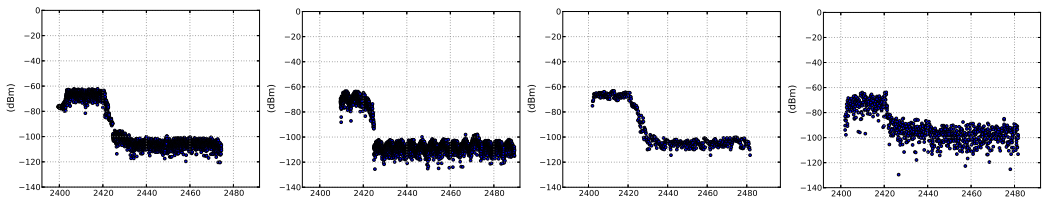
$$E(Dt) = \frac{V_{IN}}{R} \int_{t_0}^{t_1} V_R(t)dt \tag{17}$$

However it should be made clear that through the voltage sampling equipment, only a finite number of samples of $V_R(\cdot)$ are acquired over $[t_0, t_1]$ at discrete time instances.

In order to accurately measure the voltage drop across the resistors that are attached in series with the power supply of each device, we use the NITOS ACM card that is presented in Fig. 63a. The developed card supports the high sampling rate of 63 KHz and features up to three input channels, thus providing for online power consumption monitoring at both the sensing device and the attached host machine, in a joint way.

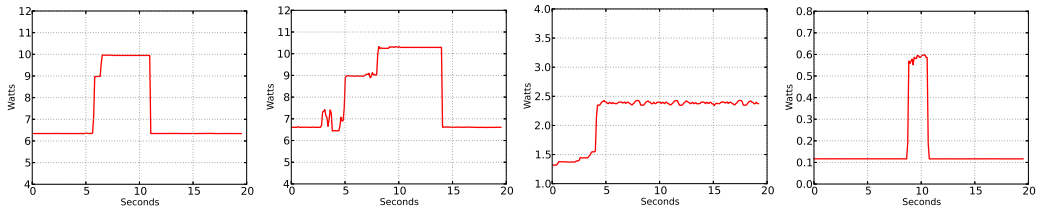
6.4.3 *Sensing Delay Distribution*

Overall sensing delay is composed of various parts that result due to different components of each device. As soon as all devices are configured properly, the first phase of actual sampling is executed. This duration does not directly correspond to the time of channel monitoring, but may also include time spent due to communication with specific device components, such as the transferring part. Third, as the measurements acquisition has been completed successfully, the next step is to process the collected measurements. Duration of this task is mainly dependent on the processing capabilities of each sensing solution. Finally, in the case that the dimension of multiple channels



(a) USRP N210 (b) USRP E110 (c) imec SE (d) Atheros AR9380

Figure 49: Power Spectral Density evaluation - Scenario 2



(a) USRP N210 (b) USRP E110 (c) imec SE (d) Atheros AR9380

Figure 50: Power Consumption evaluation - Scenario 2

is considered, the overall sensing delay may also be affected by the channel switching delay.

Ideally, the exact amount of time that is spent in each subprocess should be calculated for each device. We managed to derive the Sensing Delay Distribution for the USRP devices, by incorporating high precision timer functions at appropriate parts of the UHD source code. However, such low level measurements cannot be reported for the imec SE and the AR9380 devices, where the different subprocesses run in hardware. For these devices, we are only able to measure the total induced sensing delay and compare it with the duration of the actual sampling phase.

6.5 EXPERIMENTAL EVALUATION

6.5.1 *Power Spectral Density*

Having configured the experimental scenarios and fixed the MCD interval, we next configure the transmitter 802.11n node to constantly transmit saturated traffic on channel 6 of the 2.4 GHz band and more specifically on 2.437 GHz. In the first scenario, each device collects the fixed number of samples, as specified in Table 7. We then proceed by processing data produced by the USRP devices through FFT software, while the imec SE and AR9380 perform these calculations in dedicated hardware. For the USRPs, we use the number of samples that are closer to the next power of 2 and more specifically

process 1024 samples to produce 1024-bin FFT results for the USRP N210 and 256-bin FFT for the USRP E110. As the number of samples that are collected from the imec SE and AR9380 are more than required for a single FFT calculation, the imec SE reports the maximum values that have been collected over 10 result sets, while for the AR9380 6 different result sets are generated.

Indicative screenshots representing the PSD evaluation of the channel as monitored by each different device for the first scenario are plotted in Figures 47a, 47b, 47c and 47d accordingly. We clearly observe how the maximum bandwidth and FFT-bin size specifications of each device affect the sensing efficiency. The imec SE along with the Atheros AR9380 device detect that the 20 MHz bandwidth are fully utilized, while the limited USRP E110 is able to monitor only 5 MHz of bandwidth. On the other hand, the USRP N210 is able to detect the drop at frequencies that exceed the central frequency by 10 MHz and thus is the only device able to characterize that the Wi-Fi transmission has the central frequency of 2432 MHz (channel 5). Moreover, taking into account the increased 1024 bin FFT resolution (24.41 KHz/bin) that is generated by the USRP N210 and plotted in Fig. 47a, we observe that the PSD distribution is presented at the OFDM subcarrier level, among the 52 subcarriers that the 802.11n compatible transmission uses.

Figures 49a, 49b, 49c and 49d plot the PSD, as evaluated by each device in the second scenario. We observe that all devices detect channel 1 (freq. 2412 MHz) to be busy, while the rest part of the spectrum in the 2.4 GHz is characterized by Energy levels that are close to the Noise level (-100 dBm). In addition, the amount of channel switches that each device needs to perform in order to sense the whole 80 MHz band can also be extracted from the corresponding figures. Based on the bandwidth specification of each device, the USRP N210 (25 MHz) needs to switch its operational frequency 3 times, the USRP E110 (5 MHz) 20 times, the imec SE (20 MHz) 3 times and the AR9380 (20 MHz) 3 times, to sense the whole 80 MHz wide 2.4 GHz band.

6.5.2 *Power Consumption*

6.5.2.1 *Consumption during Measurement Collection*

In parallel with the PSD evaluation, we attach a dedicated NITOS ACM cards to each one of the devices, in order to measure the energy consumption during the spectrum sensing procedure and plot the collected results in Figures 3 and 5 for each scenario. As soon as the devices have been properly configured and the actual sensing procedure

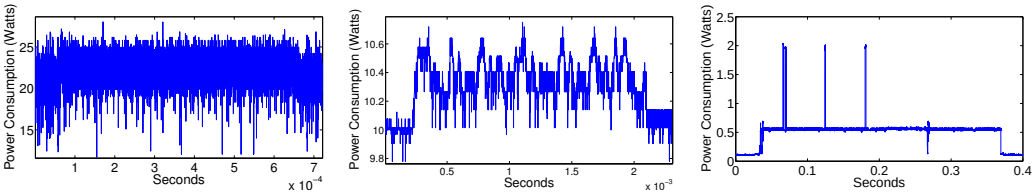
takes place, the power consumption of each device reaches a stable level, which equals 9.68 W for the USRP N210, 10.54 W for the USRP E110, 2.39W for the imec SE and 0.57 W for the AR9380 accordingly. In general, we remark that the commercial Atheros AR9380 Wi-Fi chipset presents the best energy efficiency, while the embedded imec SE is also characterized by a low power consumption profile, in comparison with the energy harvesting research oriented USRP devices. In the second scenario the power consumption presents exactly the same results, however the duration of the sensing, transferring and processing procedures is prolonged and thus the energy consumption is increased.

6.5.2.2 *Consumption during Measurement Processing*

Both the imec SE and the AR9380 devices process measurements in hardware and as a result isolation of consumption due to processing cannot be obtained. Regarding the consumption of the USRP devices N210 and E110, we measure the consumption of the ATOM-based host machine and the embedded processor, during measurements processing. Figures 51a and 51b depict the power consumption increase that the processing results in, which equals approximately 1.35 W for the ATOM and 0.4 W for the E110. In addition, Fig. 51c plots the power consumption of the AR9380 in the second scenario, where we observe spikes that correspond to the channel switching operation that lasts 1.2 ms and verifies the reported in the official product specifications channel switching delay.

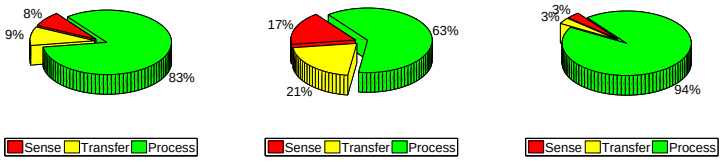
6.5.3 *Sensing Delay Distribution*

As we are not able to estimate the distribution of Sensing Delay for the imec SE and the AR9380 devices, here we present results that were obtained while experimenting with the USRP devices. Having configured the MCD interval of 64 μs, the 1600 samples that have been collected by the USRP N210 require 79 μs to be transferred over the GbE, while the USRP E110 requires 58 μs to transfer the 320 samples from the FPGA to the RAM memory of the embedded machine. Regarding the processing duration, the 1024-bin FFT operation is executed in 701 μs on the ATOM based host machine and 240 μs to run on the i7 processor. Finally, the ARM Cortex-A8 of the USRP E110 requires 1800 μs to execute the 256-bin FFT operation. Figures 52a, 52b and 52c plot the distribution of total Sensing Delay in the three different setups. We clearly notice that the processing process dominates the total delay, as it exceeds the duration of the other two phases and thus real-time sensing cannot be performed under these con-



(a) USRP N210 - ATOM (b) USRP E110 (c) AR9380

Figure 51: Power Consumption during processing of measurements collected over the MCD interval



(a) USRP N210 - ATOM (b) USRP N210 - i7 (c) USRP E110

Figure 52: Sensing Delay distribution

figurations. In USRP N210, data transfer happens in parallel with the sensing, as two different hardware modules inside the FPGA work in parallel. Hence, the reason that consists our solution incapable of performing real-time sensing is the high processing delay. However, as experimentation with longer MCD intervals, showed that the processing duration becomes significantly lower, we expect that real-time sensing can be performed in systems able to achieve significant amount of parallel processing.

6.6 EXTERNAL CASE EVALUATION

In this section, we assess the performance of an external use case that implements real-time spectrum sensing on SDRs, through a multi-threaded sensing engine software that achieves parallel processing. The sensing engine under evaluation was presented in [107] and is composed of two main threads, among which the first one is responsible for collecting samples from the USRP, while the second thread is responsible for processing the acquired samples. The sample-processing process again generates several sub-threads to process the incoming samples in parallel, by calculating the FFT-based PSD and the energy for specified channels. Once all sub-threads finish processing, they terminate and the original sample-processing thread outputs the result. We configure the sensing engine to use 16 processing threads that process in parallel the samples received from the sample-collecting thread at the highest configurable sampling rate of the USRP N210 (25 MSps). In cases that the sample-processing thread can no longer

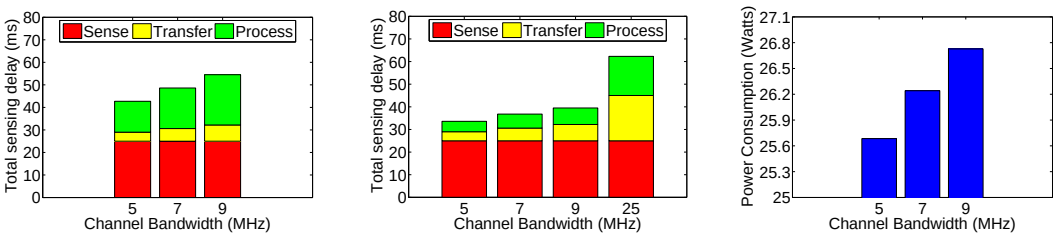
Sensing Delay	Channel Bandwidth			
Distribution	5 MHz	7 MHz	9 MHz	25 MHz
Collected Samples	125K	175K	225K	625K
Sensing	25 ms	25 ms	25 ms	25 ms
Transferring	4ms	5.6 ms	7.2 ms	20 ms
ATOM Processing	13.7 ms	18 ms	22.3 ms	-
i7 Processing	4.6 ms	6.2 ms	7.3 ms	17.3 ms

Table 8: Sensing Delay Distribution across varying bandwidth configurations for the USRP N210 when attached with ATOM and i7-based setup

follow the sample-collecting thread, the software detects the overflow of samples and terminates.

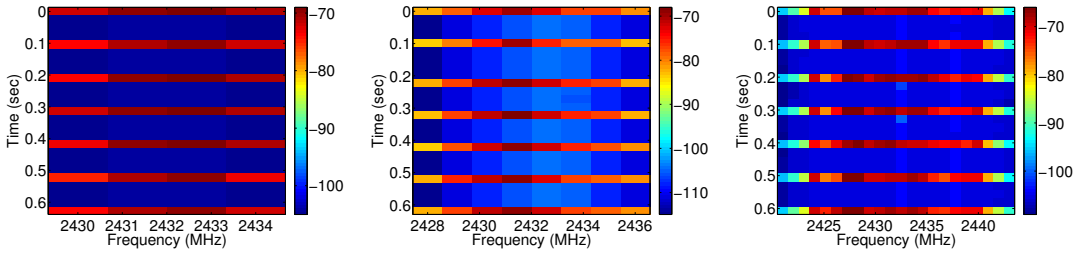
The sensing engine under evaluation has been shown (in [107]) to sense in real-time when running on a hexacore server machine. In this experiment, we use the developed framework to evaluate its performance under the dual-core ATOM-based and the quad-core i7-based setups, trying to investigate whether these setups are able to provide for real-time spectrum sensing. We configure the sensing engine to sense Channel 5 (2432 MHz) of the 2.4 GHz band and detect the presence of an 802.11 beacon that is transmitted every 100 ms. This well-defined signal is convenient for validating the performance of the channel occupancy monitoring procedure. According to the IEEE 802.11 standard [1], beacon frames are 134 bytes long and transmitted at 6 Mbps bitrate, their transmission requires approximately 179 μs. We also configure the MCD interval at 25 ms and are able to take 4 decisions about channel occupancy within the 100 ms Beacon interval.

While experimenting with the ATOM-based setup, overflow of samples was detected when the channel bandwidth was configured above the 9 MHz. On the other hand, the i7-based machine did not experience any overflows even in the 25 MHz bandwidth configuration. Figures 53a and 53b present the outcomes of our evaluation in terms Sensing Delay Distribution, across varying channel bandwidth configurations. The obtained measurements are detailed in 8 and show that both systems are capable of sensing the spectrum in real time up to the 9 MHz and 25 MHz bandwidth configuration accordingly. Under this setup, the multithreaded system implementation is able to execute the processing phase in parallel with the sample acquisition phase, as long as the processing delay is constantly lower than the sensing delay. In Fig. 53c, we also plot the average power consumption that the ATOM-based setup results in, under the three configured bandwidth values. We clearly observe that the increasing amount of data that are being processed impacts instantaneous power consumption and more specifically remark that the 5 MHz, 7 MHz and 9 MHz, resulted in the 25.684 W, 26.2413 W and 26.7276 W of average power consumption accordingly. In order to demonstrate the seamless cap-



(a) Sensing Delay Distribution of ATOM based node (b) Sensing Delay Distribution of i7 based node (c) Average Power Consumption of ATOM based node

Figure 53: External use case evaluation across varying bandwidth configurations



(a) ATOM based setup - 5 MHz (b) ATOM based setup - 9 MHz (c) i7 based setup- 25 MHz

Figure 54: Spectrogram of captured 802.11 beacon signal across varying bandwidth configurations

turing of 802.11 beacons, we plot in Figures 54a, 54b and 54c spectrograms as resulting under each setup accordingly. During 700 ms, seven beacon signals are transmitted and they are all captured, clearly demonstrating the advantage of seamless capturing.

6.7 CONCLUSIONS

In this paper we demonstrate a framework that enables evaluation of cognitive devices, in terms of sensing delay and resulting energy consumption. We present a monitoring procedure that has been directly integrated in the experimentation tools of the CREW facilities and demonstrate how it aids in the online evaluation of 4 different cognitive platforms in terms of the aforementioned metrics. We also use the developed framework to experimentally validate a real-time spectrum sensing solution that implements parallel processing on software-defined radios.

7

AGILE: ENABLING DISTRIBUTED SPECTRUM ADAPTATION IN COMMERCIAL WIRELESS LANS

7.1 INTRODUCTION

The wide adoption of wireless networking technologies in everyday life scenarios has resulted in congestion of the limited spectrum that is available in unlicensed frequency bands. The unprecedented spectrum scarcity is mainly due to the tremendous growth of IEEE 802.11 WLAN deployments. In this context, a large body of work [108, 109, 110, 111] has proposed approaches that dynamically tune the operating channel, towards avoiding heavily utilised spectrum fragments and improving throughput performance. However, as the unlicensed spectrum is also home for other wireless protocols (e.g., Bluetooth, Zigbee) and a large range of RF devices (e.g. cordless phones, security cameras, baby monitors), the channel selection process is further complicated due to cross-technology interference. While several recent works [112, 113] have enabled the identification of the exact type of interfering RF devices, the core problem of cross-technology interference mitigation through spectrum adaptation has not yet been addressed.

Identification of interference-free channels is getting even more challenging in deployments supporting the latest wireless standard versions, which adopt channels of wider width in an effort to increase the achievable data rates. More specifically, IEEE 802.11n [114] supports up to 40 MHz channels and the IEEE 802.11ac [115] further increases the channel width up to 160 MHz. Experimental measurements have verified that the throughput performance of high SNR links is nearly linearly dependent on the configured bandwidth, while this property no longer holds in the presence of interference. Towards alleviating the identified performance discrepancy, a great deal of research literature [116, 117, 118, 119, 120, 121] has advocated sophisticated mechanisms that dynamically adapt the occupied spectrum to the prevailing channel conditions. However, most of the aforementioned approaches were not compatible with the 802.11, thus not offering for direct integration and comparative performance evaluation with the standard. On the other hand, the work in [116] was the first to implement and evaluate variable bandwidth configurations on a single, isolated 802.11 link, while FLUID

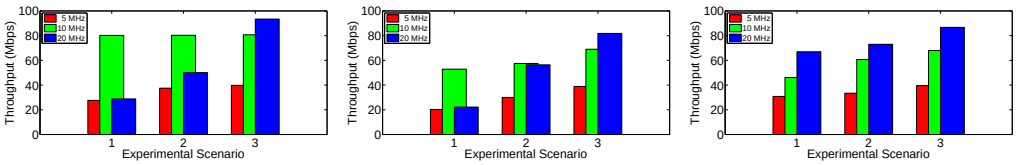
[121] presented the first large-scale evaluation of flexible spectrum configurations in enterprise 802.11 WLANs. In the case of FLUID, a central controller entity is used to adapt spectrum configurations per link and resolve potential conflicts, thus consisting the approach applicable only to centrally managed WLAN deployments.

In this work, we present the AGILE system that is able to configure both the central frequency and bandwidth of operation on an per link basis. Identification of under-utilised spectrum fragments is based on spectral analysis of measurements obtained at the PHY-layer through hardware that is inherent in standard OFDM receiver circuits of commercial 802.11 hardware. Next, considering the application-layer requirements of each link, we assign the minimum bandwidth that is able to meet the Application-layer traffic requirements. AGILE presents the advantages of experiencing increased SNR and potentially greater throughput under interfered conditions and decreased power consumption, as presented in [116]. Moreover, higher spectrum efficiency is obtained, as more part of the spectrum is available for use by collocated wireless links, while the link under consideration still achieves the required performance.

In the rest of this chapter, we start by executing extensive sets of experiments to characterise the performance of links that operate on flexible operating frequencies and bandwidth configurations. Experimental evaluation considers both the metrics of throughput performance and energy efficiency. Next, we present the detailed design of the sensing engine and show how it is able to detect interfering transmissions as generated by several types of non-802.11 RF devices. We also present the principle on top of which we classify the impact that each type of interfering transmitter results in, towards appropriately adapting the operational parameters when multiple interfering links are concurrently activated. Finally, we analyse the mechanism that decides about the appropriate central frequency and channel bandwidth, by taking into account both the application-layer requirements along with the prevailing channel and traffic conditions.

7.2 EXPERIMENTAL EVALUATION

We observed that, in isolation, the throughput for high SNR links nearly doubles on doubling the channel width, as expected according to Shannon's channel capacity theorem. However, in the presence of even one interferer, this property no longer holds. In this initial set of experiments, we aim at investigating the potential of our system to adapt under rich in interference environments. To this end, we establish two wireless links on the 5 GHz band in an office environment and use them as potential interferers for a third link that is able to configure both its central frequency, as well as its channel



(a) Topology 1 (b) Topology 2 (c) Topology 3

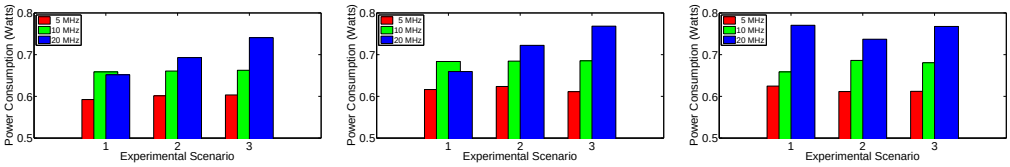
Figure 55: Throughput Performance evaluation.

bandwidth. In the first scenario, the two interfering links use the bandwidth of 20 MHz and operate on frequencies 5221 MHz and 5251 MHz accordingly, leaving a part of 10 MHz spectrum space unoccupied. In the second scenario, we switch the central frequency of the second interfering link by 5 MHz and assign the frequency of 5256 MHz, while in the third scenario the frequency of 5261 MHz is assigned. Following this setup, the non-occupied part of the spectrum equals 15 MHz and 20 MHz in the second and third phase.

Towards deriving the optimal frequency and bandwidth configuration under each setup, we evaluate the performance of the third link, while configuring all the available frequency configurations between the 5221 MHz and 5251 MHz and its bandwidth between the 5 MHz, 10 MHz and 20 MHz values. We test this setup under three different topologies and as the full set of configurations is rather extended, we decided to present a subset of indicative measurements, in order to save space. More specifically, we present results obtained under setups, which use central frequencies that do not cause any degradation to the throughput performance of the interfering links when the 5 MHz and 10 MHz bandwidths are configured. More specifically, for the first topology, we fix the identical central frequencies of 5236 MHz, 5237 MHz and 5239 MHz for all the considered bandwidths in each scenario accordingly.

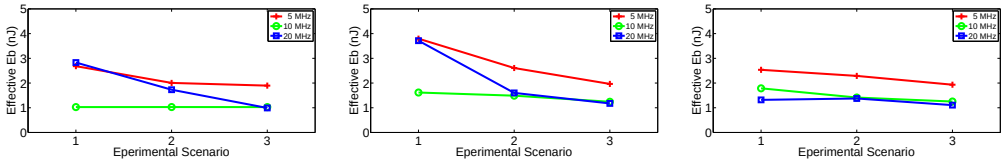
7.2.1 *Throughput Performance*

Figures 55a, 55b and 55c present the throughput performance, as obtained under each bandwidth configuration for each considered topology. Considering the first topology, we observe that the throughput performance of the 5 MHz and 20 MHz setups is improving across increasing separation of central frequencies. In the third scenario, we notice that the 5 MHz configuration reaches the maximum performance of 39 Mbps, while the 20 MHz setup is only able to achieve 93.4 Mbps, which is far less than the maximum achievable performance of approximately 160 Mbps for the 20 MHz bandwidth. We also observe that the 10 MHz bandwidth setup constantly achieves approximately 80 Mbps throughput, which is also the maximum obtained performance for the first two scenarios.



(a) Topology 1 (b) Topology 2 (c) Topology 3

Figure 56: Power Consumption evaluation.



(a) Topology 1 (b) Topology 2 (c) Topology 3

Figure 57: Energy Efficiency evaluation.

In the second topology, we configure a setup where the three links are dissociated, thus minimizing the impact of interfering links on the link under consideration. The same identical central frequencies of 5236 MHz, 5237 MHz and 5239 MHz are again fixed for all the considered bandwidths in each scenario accordingly. Again we observe that the 10 MHz setup achieves the maximum performance in the first two scenarios, while only in the third scenario the 20 MHz configuration outperforms the throughput obtained by the 10 MHz setup. In the third topology, we not only isolate the three links, but also place the intended receiver of each link at an increased distance, thus reducing the RSSI at each receiver. In this topology, we configure the identical central frequencies of 5238 MHz, 5241 MHz and 5241 MHz for all the considered bandwidths in each scenario accordingly. We observe that the 5 MHz configuration achieves the highest throughput across the rest bandwidth setups, when considered proportionally to its achievable Physical bitrate. Considering the 10 MHz and 20 MHz setups, we remark that their performance is improved as the separation with the central frequencies of interferers is increased, while the maximum achievable throughput is not obtained even at the maximum considered separation.

7.2.2 *Power Consumption Performance*

We also measure the instantaneous power consumption at the receiver of the third link for each bandwidth configuration and across the three topologies. In order accomplish this, we configure the experimental setup and use the NITOS ACM hardware that is described in Chapter 8. In Figures 56a, 56b and 56c, we present the average power consumption as obtained during the execution of each unique experiment. The figures indicate a linear relationship between the channel width and the power consumption.

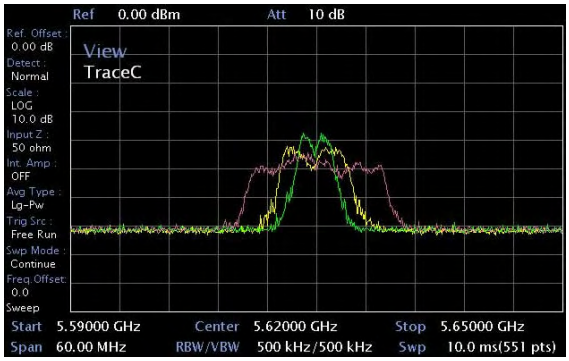
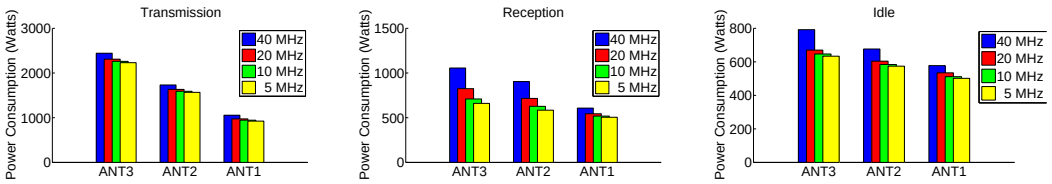


Figure 58: Indicative Screenshot of the spectrum analyzer showing 20MHz, 10MHz and 5MHz signals

We clearly notice that wider channels consume more power. The only exception to this observation is related to the performance of the 20 MHz setup in the first two topologies, in which cases the link rarely receives frames and thus remains in the idle state for longer duration, thus resulting in lower average power consumption. The decrease in power consumption can be explained by a slower clock speed that is used at narrower channel widths. We present detailed power consumption measurements in the next section, while for the rest of this section we focus on evaluating the Energy efficiency per received bit of information (E_b), in order to derive the most energy efficient mode of operation for each setup.

Figures 57a, 57b and 57c illustrate the effective E_b, as calculated proportionally to the throughput performance that each setup is able to achieve. We clearly observe that the 20 MHz bandwidth configuration proves to be the most energy efficient mode, as it achieves the lowest effective E_b across nearly all the tested setups. This observation comes due to both its increased throughput along with the lower average power consumption that is results in. We also observe that the 5 MHz configuration is found to be the least energy efficient. The prolonged reception duration, due to the resulting decreased Physical bitrate is not able to counterbalance the reduced average power consumption.

Concluding, we remark that under low-SNR conditions, lower bandwidth configurations are able to provide increased throughput performance. Moreover, we observed that the frequency separation threshold (FS_{THR}) of 20 MHz is not able to isolate links of 20 MHz bandwidth, while even the FS_{THR} of 10 MHz resulted in isolated transmissions and high-SNR conditions for the 5 MHz and 10 MHz bandwidths. This fact might come due to the increased by 3 dB SNR that halved bandwidth transmissions result in, as depicted in Fig. 58. Another important outcome of this experiment is that isolation of lower bandwidth links can be established with a higher flexibility. However, we did not observe similar behaviour for the 5 MHz



(a) Transmission (b) Reception (c) Idle

Figure 59: Power consumption across varying Bandwidth configurations

setup in the first two topologies. Towards arriving at safe conclusions, we plan to investigate the performance of links that use different bandwidths in several topologies.

Considering the power consumption measurements, we observe that huge potential exists for increasing the energy efficiency under low-SNR conditions through bandwidth adaptation. In cases that the obtained throughput performance is increased by lowering the channel bandwidth, the energy efficiency is further more increased, as a result of the lower power consumption that lower bandwidths are able to provide on average. The above observations clearly highlight the potential for increasing performance both in terms of throughput and energy efficiency performance by adapting the the frequency and bandwidth configurations. Moreover, they further motivate us to investigate how the optimal configurations can be derived under varying channel and topology conditions.

7.2.2.1 *Detailed Power Consumption Profiling*

Figures 59a, 59b and 59c present the instantaneous power consumption under different channel bandwidths while operating transmission, reception and idle mode accordingly. We present results for the highest achievable Physical bitrate per antenna configuration, while there were not monitored variations across different modulations. The figures indicate a linear relationship between the channel width and the power consumption and also clearly show that wider channels consume more power. Detailed measurements are listed in Table 9 for the 3x3 MIMO antenna configuration. These results show that configuration of 5 MHz bandwidth is able to provide savings of 9%, 38% and 21% in comparison with the 40 MHz configuration, when transmitting, receiving and idling accordingly. Our detailed measurements interpret the significant energy savings that were attained by using lower bandwidth configurations in the initial sets of experiments.

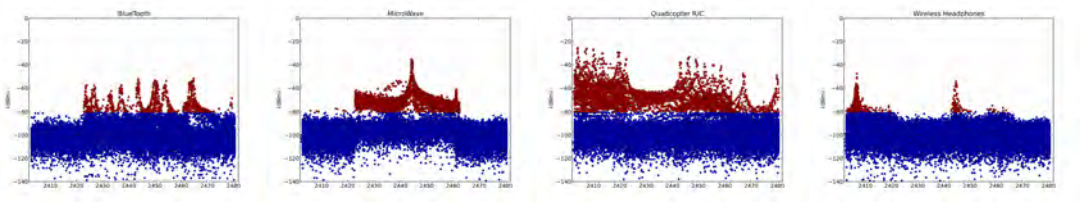
Mode	TX	RX	IDLE
Bandwidth	Power Consumption (Watts)		
40	2.442	1.056	0.792
20	2.31	0.825	0.669
10	2.2605	0.795	0.646
5	2.2308	0.66	0.633

Table 9: Power consumption across different operational modes and bandwidth - 3x3

7.3 SPECTRUM SCANNING MECHANISM

Traditionally, WiFi cards have not exposed their inherent spectral scanning capabilities. In this work, we use the commercial Atheros AR9380 chipset that provides access to by the spectral samples that are the output of the underlying FFT hardware at the PHY layer. We use the card in 802.11n 20 MHz HT mode, where a 20 MHz channel is divided into 64 sub-carriers, spaced 312.5 KHz apart and the signal data is transmitted on 56 of these sub-carriers. Each spectral sample (FFT) generated by the wireless card comprises the power received in 56 sub-carriers (FFT bins) and corresponds to a 17.5 MHz (56 x 0.3125 MHz) chunk of spectrum, which we refer to as a sub-channel. We aim at exploiting this information in order to evaluate the Spectrum Occupancy on an energy detection level, so as to be able to detect transmissions that have been generated by non-WiFi devices, as well as by transmissions that use a different bandwidth configuration that are no longer decodable. The output of the Spectrum Scanning mechanism will then be fed to the mechanism responsible for selecting the optimal operational frequency and bandwidth configurations.

As the Spectrum Occupancy evaluation mechanism should be able to detect different types of transmissions, our occupancy evaluation metric should be rather generic. We start by characterising the PSD as impacted by transmissions of different types of RF hardware. More specifically, we use the Bluetooth 4.0 transceiver of a Google Nexus 4 Smartphone, a Microwave oven, a wireless headset and a Quadcopter R/C unit. The PSD as captured by our sensing mechanism, during the operation of each different device is plotted in Figures 60a, 60b, 60c and 60d accordingly. The Bluetooth transceiver uses 1 MHz of bandwidth and hops to different frequencies every 625 μs, while the Quadcopter R/C unit also performs frequency hopping. Several studies have confirmed that Bluettoth is not able to significantly impact the performance of 802.11 links, due to it relatively low transmission power of 10 mW. On the other hand, the Quadcopter R/C is able uses a relatively high Transmission Power that exceeds the 200 mW limit of commercial Wi-Fi cards and thus is considered as a heavy interferer. The wireless headset uses a limited amount of bandwidth on two central frequencies that



(a) Bluetooth 4.0 (b) Microwave oven (c) Quadcopter R/C unit (d) Wireless headphones

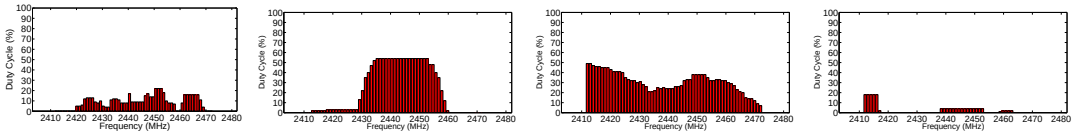
Figure 60: PSD as evaluated during the operation of each different RF device

are frequently switched when interference is detected. Also its transmission power is relatively low, estimated close to the output power of the Bluetooth transmitter. Finally, the microwave oven, which typically emits high RF energy in 2.44-2.47 GHz frequencies, has a duty cycle of 0.5, resulting in an ON-OFF pattern, typically periodic with a period of 16.66 ms.

Having experimented with varying types of RF transmitters, we observed that a clear relationship exists between the transmission duty cycle that each device uses and its impact of 802.11 transmissions. This fact is directly related to the operation of the CSMA channel access mechanism of the 802.11 protocol. As another transmission is detected to exceed the CSA threshold, any 802.11 compatible card refrains from transmitting and back-offs till it again tries to access the medium. In cases that the channel is detected to be clean, the 802.11 compatible transmitter is able to capture the channel. On top of these observations, we start by characterising the duty cycle as observed per each MHz of the available set of configurable frequencies. There is a huge amount of works related with the definition of appropriate energy detection thresholds. However, as our system follows the standard 802.11 CSMA implementation, we plan to use the thresholds specified by the standard. 802.11 defines the CCA threshold to be equal to -82 dBm for 20 MHz channel spacing, -85 dBm for 10 MHz channel spacing, and -88 dBm for 5 MHz channel bandwidth. However, in the case that the preamble is missed, the receiver detects the medium as busy for any signal 20 dB above the minimum CCA threshold. As the proposed system supports different bandwidth configurations, we will use a different duty cycle detection threshold (DC_{THR})to detect which transmissions would potentially impact the link under consideration. We plan to experiment with varying threshold values and under varying topologies, in order to detect the optimal value that ensures successful detection of interfering links.

7.4 OPTIMAL FREQUENCY DETECTION

Spectral scanning enables the detection of any type of RF transmission and base on the Duty Cycle classification of detected signals is able to identify “Spectrum Holes",



(a) Bluetooth 4.0 (b) Microwave oven (c) Quadcopter R/C unit (d) Wireless headphones

Figure 61: Duty cycle as evaluated during the operation of each different RF device

referring to portions of the spectrum that are not utilised in the time interval under consideration. Our solution aims at identifying “Spectrum Holes" and properly configuring the appropriate bandwidth that provides for uninterrupted transmissions.

Given as input the spectral analysis, our mechanism collects information about the measured Energy level per frequency bin (312.5 KHz). Next, the duty cycle detection mechanism considers every single central frequency F_c, across the 73 available frequency configurations in the 2.4 GHz band and evaluates the Duty Cycle, by considering all frequency bins that exist between the F_c - $BW/2$ and F_c + $BW/2$ part of the spectrum. The available BW levels that are taken into account in each step include the 5 MHz. 10 MHz, 20 MHz and 40 MHz values. For any given central frequency F_c we characterise the duty cycle of the spectrum part between the F_c - $BW/2$ and F_c + $BW/2$ to be occupied if the Energy level of any considered frequency bin is measured to exceed the DC_{THR}. Evaluating every single frequency of the available spectrum, we are able to find the least utilised part of the spectrum of BW width.

Figures 61a, 61b, 61c and 61d evaluate the channel occupancy when each different type of considered transmitters is activated, by using the DC_{THR} of -80 dBm and considering the BW configuration of 5 MHz. Although the selection of the -80 dBm DC_{THR} value is rather generic, we observe that is capable of distinguishing between heavy and light interferers. In addition, we clearly observe that for any given type of interferer a specific part of the spectrum can be identified as under-utilised. Taking the microwave oven as an example, we see that the part of the spectrum between frequencies 2410 - 2430 MHz and 2460 -2480 MHz is detected to be low utilised and not utilised at all accordingly.

In order to test this configuration, we configure an 802.11n link to operate on F_c 2437 MHz and to use 20 MHz of bandwidth. This link is able to achieve 65 Mbps when the microwave oven in not activated, while its throughput is halved when the Microwave oven is turned ON. On the other hand, the performance of this link is no longer affected when the central frequency is switched to 2479 MHz that is characterised as a “Spectrum Hole" by Duty Cycle detection mechanism. In addition, we also evaluate the performance of the 802.11n link in the case that the other heavy interferer Quadcopter R/C is activated. By testing all the 73 available F_c configurations, we observe that the

minimum throughput is obtained in the case that the 802.11n link operates on F_c 2412 MHz, which F_c is also characterised by the highest DC by the developed mechanism. Configuration of the F_c of 2479 MHz achieves the highest observable throughput performance, as it is detected to be occupied less frequently by the DC detection mechanism.

7.5 CONCLUSIONS AND FUTURE WORK

The aforementioned experimental results have demonstrated the feasibility of our approach and have also showcased that usage of flexible channel central frequencies and bandwidths are able to improve wireless performance, both in terms of throughput and energy efficiency. As the design of flexible channels is rather challenging, we developed a generic optimal frequency detection mechanism that is based on the classification of transmitters based on the Duty Cycle that their transmissions follow. The next step of our research is to test the developed framework across varying topologies and network configurations, in order to derive the optimal decision mechanism rules. More specifically, we need to identify generic appropriate Duty Cycle detection thresholds (DC_{THR}) and Frequency Separation thresholds (FS_{THR}) that can drive optimal decision of optimal central frequencies per each different bandwidth configuration.

Part III

ENERGY EFFICIENCY

8

REAL-TIME POWER MONITORING IN EXPERIMENTAL WIRELESS NETWORK DEPLOYMENTS

8.1 INTRODUCTION

The unprecedented penetration of "smart" mobile devices in everyday use cases, has greatly affected the trends followed by vendors developing such equipment. First, the need for offering ubiquitous network coverage, has led the industry in equipping these devices with several wireless interfaces (WiFi, Bluetooth, 3G, LTE, WiMAX), to facilitate parallel network access. Second, in an effort to meet the increasing requirements generated by the use of resource-demanding applications, high-end mobile devices feature multi-core processors, high-resolution displays and support increased data rate communication technologies. Especially in the case of smartphone platforms, the energy greedy profile of the supported state-of-the-art wireless technologies may induce up to 50% of the total platform power consumption [6], under typical use case scenarios. The increased energy demands of such technologies cannot be successfully met, due to the limited energy capacity [7] that existing battery technologies are able to offer.

The overall goal, towards alleviating this unique performance discrepancy, is to reduce energy consumption wherever possible. Towards this goal, several recent research studies [122, 123, 124, 125] in the field of wireless networking have focused on reducing the total amount of energy consumed during the wireless medium access and communication operations. In this context, accurate energy consumption assessment needs to be applied by the research community, as a means of evaluating the energy efficiency of the proposed protocols and architectures. Researchers working on wireless sensor networks can base their evaluations on detailed low level specifications [126, 127] provided by developers of widely adopted sensor platforms (Tmote Sky, MICAz). However, this is not the case with vendors that develop wireless transceivers for everyday use devices, such as laptops, smartphones or tablets, where only limited information on nominal consumption is publicly provided [105]. Even in cases that accurate data sheet specifications are available, power consumption models that are based on such ac-

curate measurements, fail to successfully calculate energy expenditure under complex configurations and topologies.

In order to enable experimenters to accurately evaluate the energy efficiency of the proposed protocols, under real world scale and settings, advanced methodologies and solutions need to be developed. In this work, we propose the innovative NITOS EMF framework that is fully integrated with the large-scale wireless NITOS testbed [80] and provides for online gathering of energy measurements, through a distributed network of low-cost, but highly accurate devices.

8.1.1 *Related Work*

As previously stated, a great variety of research efforts has proposed energy-efficient protocols and architectures, towards moving to "greener" solutions in telecommunications. Mechanisms proposed in recent works range from scheduling of sleep intervals and antenna configurations [128], to reduction of time spent during idle listening periods [122] and application of adaptive transmit power and physical layer (PHY) rate control [123]. The approaches above, jointly follow the methodology of first identifying key functions that exacerbate energy expenditure and subsequently attempt to control the induced impact by efficiently scheduling their operation. Other relevant works focus more on energy consumption characterisation of specific technologies, such as the power-hungry IEEE 802.11n [129], or specific platforms, such as the energy-limited smartphones [124, 125]. Trying to address the limited level of detail provided by the industry, several works [130, 6] present extensive measurements that assess the impact of low-level configurations on the overall power consumption of various platforms. In order to gather such detailed measurements, either commercial sensing hardware or custom measurement setups are employed, which result in varying levels of obtained accuracy and reliability.

In an effort to support evaluation under definite measurement setups, a limited variety of approaches have developed proper energy consumption monitoring frameworks for testbeds that specifically target wireless sensor networks [131] or data centers [132]. Similarly, the work in [133] proposes a framework developed for energy consumption monitoring of real WLAN deployments. However, this framework is restricted in characterising the consumption at the level of complete device (router). The importance of monitoring consumption of both the wireless network interface card (NIC) and the complete device as a total, in deriving hitherto unexploited tradeoffs, is highlighted in [**?**]. Based on comparison of energy consumed at the transceiver and the complete

device level, this work proposes a novel energy model that contrasts traditional approaches, which neglect the fraction of energy consumed while each frame crosses the protocol stack.

This paper is organized as follows. In sections 2,3 and 4, we present the required platform specifications, the followed measurement methodology and detail the various framework components accordingly. In section 5, we present extensive experiments that evaluate the platform's capabilities, while in sections 6 and 7 we discuss our future work and point out conclusions accordingly.

8.2 PLATFORM SPECIFICATIONS

The overall goal of the proposed work is to turn the rather challenging task of online energy consumption monitoring into an automated procedure that is available to wireless testbed experimenters. Below, we list the various required platform characteristics, along with the implementation choi- ces that we followed, in order to integrate them with the developed framework:

- **Non-intrusive operation:** The normal network operation should not be affected by the monitoring procedure, in order not to result in imprecise results. Our solution runs on external hardware, which does not interfere with the measured devices.
- **Online monitoring:** Online monitoring needs to be applied, towards energy efficiency evaluation under complex configurations and topologies. The proposed framework allows for online gathering of measurements in parallel with execution of long-term experiments.
- **Distributed architecture:** Assessment of the impact induced by simultaneous operation of collocated network components, requires the development and application of distributed energy consumption monitoring solutions. The proposed framework is composed of distributed communication enabled components.
- **High sampling rate:** State-of-the-art wireless technologies are capable of high transmission rates, which generates the necessity for energy monitoring devices to feature high sampling rate components that allow monitoring of short duration events (in the order of μs). The developed platform achieves twice the minimum required sampling rate.

- **High sampling accuracy:** Wireless testbed infrastructure may operate in states that result in similar energy consumption, thus necessitating the application of high accuracy sampling equipment able to distinguish between the energy consumption induced in such operational states. The custom developed hardware achieves accuracy in the order of milliwatts.
- **Adaptive to heterogeneous infrastructure:** Due to the existence of several heterogeneous types of interfaces, protocols and architectures, connectivity of the developed solutions with the components under test should be easily setup in all cases. The followed measurement procedure is rather generic and allows for power consumption monitoring of any device type.
- **Low-cost hardware:** The extended scale and increasing scalability of modern experimental infrastructure require that the developed hardware components are of low cost to allow for the distributed deployment of several monitoring devices. The developed hardware solution introduces a total cost of less than 80.

8.3 MEASUREMENT METHODOLOGY

In order to estimate the instantaneous power consumption of any device, we follow a widely adopted power measurement procedure, which requires the placement of a high precision, low impedance current-shunt resistor (R) of a known resistance value, in series with the power source and the power supply pin of the device to be measured. The exact measurement setup described above is presented in Fig. 62.

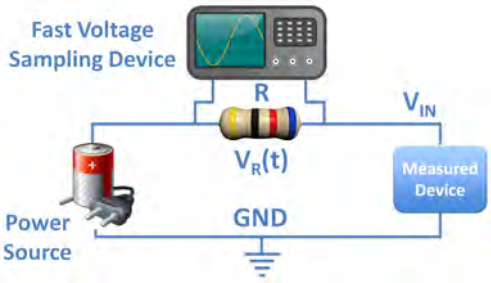
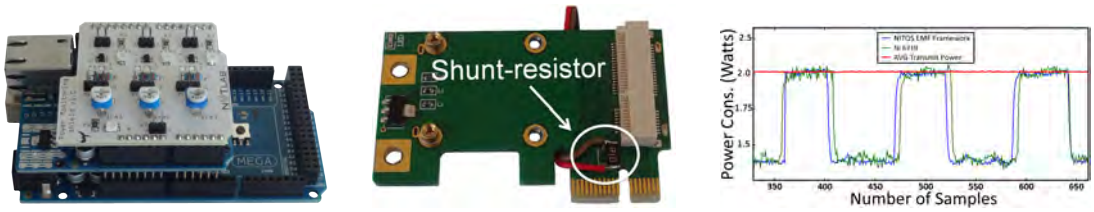


Figure 62: Representation of Power Measurement setup

By consistently measuring the voltage ($V_R(t)$) across the current-shunt resistor through proper voltage metering equipment, we are able to extract the instantaneous current draw of the device, based on Ohm's law. The instantaneous power consumption can be calculated as the product of the input voltage V_{IN} and the measured current draw:

$$P(t) = V_{IN} \frac{V_R(t)}{R} \tag{18}$$



(a) NITOS ACM card (b) Modified mPCI-e adapter (c) Python Plotting software

Figure 63: NITOS ACM card and accompanying hardware and software components

In order to estimate the total energy consumption during specific events, we first need to estimate the event's duration. Total energy consumed over an interval $[t_0,t_1]$ is the integral of power consumption over the specified time duration ($Dt = t_1 - t_0$), calculated as:

$$E(Dt) = \frac{V_{IN}}{R} \int_{t_0}^{t_1} V_R(t)dt \tag{19}$$

For instance, the duration of a single frame transmission or reception can be directly obtained as the product of the frame length and the configured PHY-layer bit rate. Subsequently, energy consumption can be obtained as the integral of the power consumption over the calculated duration. However it should be made clear that through the voltage sampling equipment, only a finite number of samples of $V_R(\cdot)$ are acquired over $[t_0, t_1]$ at discrete time instances.

8.4 NITOS EMF FRAMEWORK

NITOS EMF, which integrates both hardware and software components, manages to implement the aforementioned power measurement approach. Below, we detail the platform characteristics and describe how the framework has been integrated with the testbed architecture.

8.4.1 *Hardware Components*

The developed hardware device, is built on top of the first version of the NITOS Chassis Manager (CM) Card [134], which was initially used to control the operational status of testbed nodes. The advanced version of the card (*NITOS ACM*), which is presented in Fig. 63a, is mainly composed of Arduino compatible open-source components, but also features custom developed hardware. The various hardware components are detailed below:

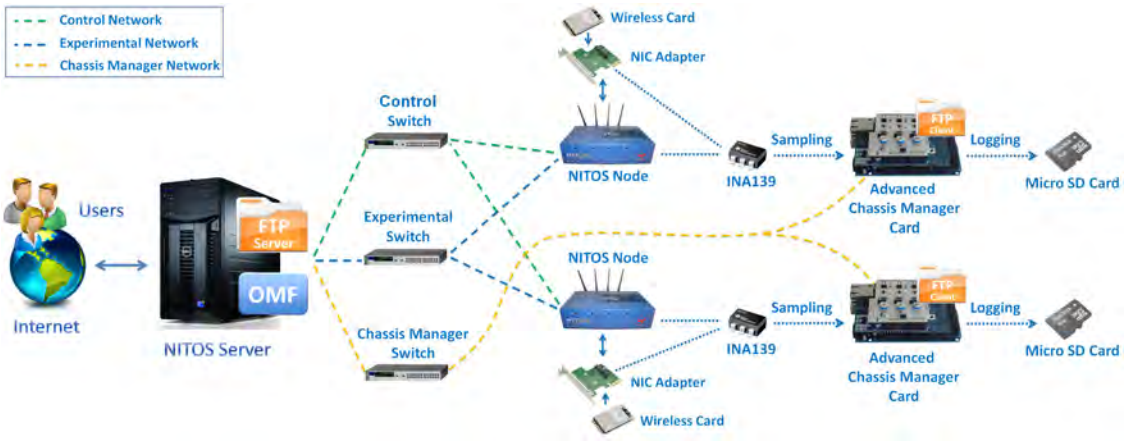


Figure 64: Integration of NITOS EMF framework with the overall testbed architecture

Arduino Mega 2560

The developed card is based on the low cost Arduino Mega 2560 board, featuring the ATmega2560 [135] 8-bit micro-controller that runs at 16 MHz and operates at 5 Volts. The ATmega2560 integrates a 16 channel Analog to Digital Converter (ADC), with a resolution of 10-bit (i.e. 1024 different values), to provide for sampling of analog signals. We use the integrated ADC to sample the voltage across the shunt resistor as presented in Fig. 62. We decided to use the Arduino Mega 2560, as it offers 256 KB of flash memory and 8 KB of SRAM, which features are required for the efficient operation of the developed software components.

Ethernet Shield with Micro SD card

In order to provide for distributed measurements, the card should feature network communication capabilities. Towards this goal, we decided to equip the card with the Arduino Ethernet Shield [136], which features the embedded Wiznet W5100 network controller that implements a network (IP) stack capable of both TCP and UDP communication. Another key characteristic of this shield, is the embedded Micro SD card module that provides the board with external storage capabilities, enabling for long-term data logging.

Custom Shield integrated with the INA139 IC

Since the integrated ADC is not able to accurately digitize the attained voltage levels on shunt resistors in cases where the monitored voltage drop is minimally varied (mV range), we equipped the developed card with the Texas Instruments INA139 [137] Integrated Circuit. INA139 is a high-side current-shunt monitor that converts a differential input voltage to an amplified value, where the amplification level is controlled through an external load resistor (R_L) and can be set from 1 to over 100. The amplification accuracy of the INA139 Integrated Circuit (IC) is directly dependent on the selection of the current-shunt (R) and load resistor (R_L)values.

In order to decide about the proper value of the shunt resistor, we have to consider the average consumption of the device that will be measured. Considering that commercial wireless NICs have an average consumption of 2 Watts we select to use a shunt resistor of 0.1 Ω, which attains 60 mV of shunt voltage that is within the specified limits [137]. Considering the configuration of the R_L resistor, we decided to use a 30 KΩ resistor, which meets the maximum output voltage requirement of 2.725 Volts that is specified in the INA139 data sheet. Having properly configured the resistor values that control the INA139, we then designed and fabricated a Printed Circuit Board (PCB), which can be directly integrated on the Arduino board. The designed PCB, which can be seen on top of the Arduino hardware in Fig. 63a, features three individual INA139 and all the required electrical components, providing for power consumption monitoring of multiple devices.

Custom mini-PCIe adapter

Having decided about the proper value of the current-shunt resistor, we next had to attach it in series with the power supply pins of several wireless NICs. In order to ease the interception of the power supply pins and refrain from modifying each different type of NIC, we followed a more applicable approach and inserted the current-shunt resistors on communication bus adapter cards. Fig. 63b illustrates a modified pci-e to mini pci-e adapter card that is attached with a high-precision current-shunt resistor of 0.1 Ω.

8.4.2 *Software Components*

Towards rendering the described hardware platform into a functional framework, we developed appropriate software to control the ACM cards and integrate them with the testbed.

Arduino Software

The sampling rate of approximately 9 KHz that is supported by the default Arduino configurations, is not sufficient for sampling events that last for time intervals in the order of μs. We consider transmissions of typical MPDU frames of 1534 bytes length as the shortest in duration event, which requires approximately 27μs to be transmitted at the TX_R of 450 Mbps. As a result, we need to achieve sampling rate higher than 37 KHz, in order to be able to monitor such transmissions.

To overcome this issue, we modified the default ATmega2560 ADC configurations that control the achievable sampling rate. First of all, we configured the ADC to op-

erate in the free-running mode, which enables the ADC to continuously perform conversions without requiring proper signalling from the μC. Through this modification, we enable SD card logging of sampled data to take place in parallel with ADC conversions, efficiently increasing the amount of time spent in sample acquisition. Moreover, we properly modified the clock speed of the ADC prescaler from the default value of 125 KHz to 1 MHz, following the approaches in [138, 139]. Finally, we efficiently exploited the available SRAM of ATmega2560 to directly store up to 13 blocks of 512 bytes buffers into the SD card. Based on the aforementioned modifications we manage to achieve the remarkably increased sampling rate of 63 KHz, with 10-bit resolution, while only reducing the perceived accuracy by approximately 11% [140]. Additionally, we modified the default ADC voltage reference to 2.56 Volts to enable conversions of higher accuracy. The developed platform has been evaluated in comparison with the high-end NI-6210 data acquisition (DAQ) module [141] and proved of providing measurements of similar accuracy in the range under consideration.

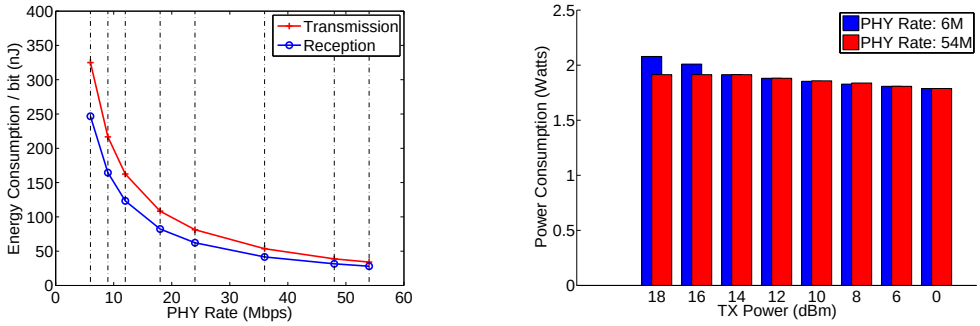
Towards providing for remote control of the distributed cards, we developed a tiny Web Server that is based on the Arduino Ethernet Library and operates on each individual ACM card. Through the transmission of custom UDP packets, we can remotely trigger the measurement acquisition procedure. Furthermore, we developed an FTP service that provides for collection of captured data in a distributed way.

Python Software

We also developed a set of Python scripts that enable direct access to the collected results and moreover precise power and energy consumption calculations. Fig. 63c presents the implemented plotting component and also depicts a comparison between measurements gathered through the developed framework and the high-end NI-6210 device. The overall set of developed software components is publicly available for users of the NITOS testbed.

Integration with OMF

To enable ease of use of the developed framework, we integrated its functionalities into the OMF cOntrol and Management Framework [26]. Based on this integration, experimenters can fully configure the operation of the ACM cards and moreover collect and access the gathered measurements through the OMF Measurement Library [51].



(a) Varying PHY-layer TX Rate (b) Varying TX Power

Figure 65: Consumption of AR5424 NIC during transmission across different configurations

8.4.3 *Framework Architecture*

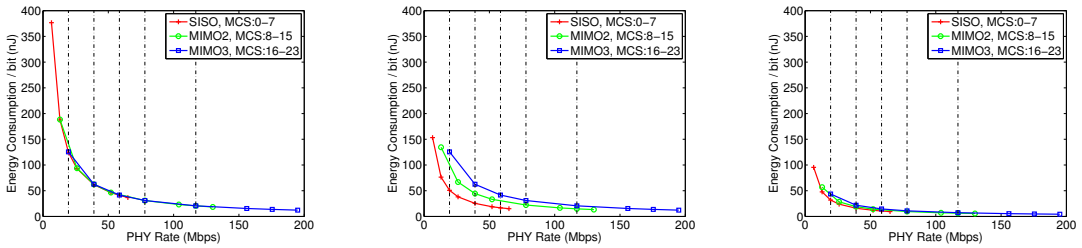
The proposed framework has been directly integrated with the underlying network architecture of the NITOS testbed.

8.4.3.1 *NITOS Testbed*

NITOS testbed currently offers 50 wireless nodes and provides open remote access to any researchers who would like to test their protocols in a real-life wireless network. The testbed architecture is illustrated in the left part of Fig. 64. Two Gigabit Ethernet switches interconnect the nodes with NITOS server, namely the *Control* switch that provide for control of experiment execution and measurement collection and the *Experimental* switch, which can be used for conducting wired experiments. A third Gigabit Ethernet, namely the *Chassis Manager* switch, is dedicated in controlling the operational status of the nodes through the transmission of custom http requests that control solid state relays on the Chassis Manager cards. NITOS nodes feature up to 3 wireless NICs, using the Atheros AR5424 and AR9380 chipsets.

8.4.3.2 *Integration of NITOS EMF with the Testbed*

Currently 20 of the nodes are attached with ACM cards, which together with the modified mini-PCIe adapters and node power supplies, enable for online energy consumption monitoring in a distributed way. The integration of the NITOS EMF framework with the overall testbed architecture is illustrated in Fig. 64. NITOS ACM cards are properly configured to monitor both the consumption of the NIC in an individual way, as well as the total consumption of each node. Through the *Chassis Manager* switch and the developed FTP service, measurements logged locally in the Micro SD



(a) Transmission - All RF chains constantly enabled

(b) Transmission - Only required number of RF chains enabled

(c) Reception - Only required number of RF chains enabled

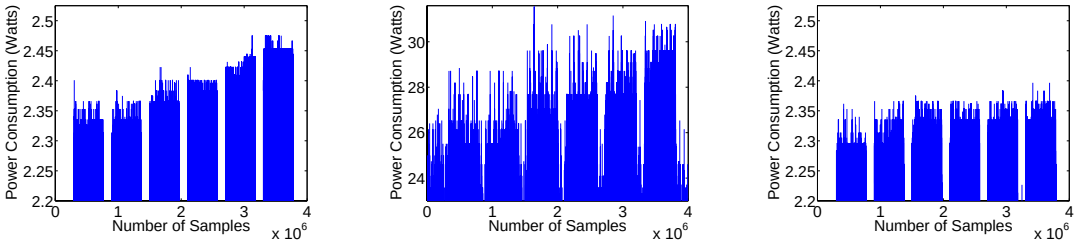
Figure 66: Energy Consumption per bit of AR9380 NIC during transmission across different MCS and Antenna settings

card module of each individual ACM card are transferred in a distributed way to the NITOS server.

8.5 EXPERIMENTAL EVALUATION

Based upon NITOS testbed that implements the proposed framework, numerous experiments were conducted, and the results obtained are analyzed in this section. The first set of experiments has been designed to demonstrate the ability of the framework to accurately monitor the effect induced by low-level configurations in the overall consumption of wireless NICs. As the range of the available low-level configurations that can highly impact energy consumption is rather extended, we detail our experimental results and findings in a technical report [142] and only present a representative sample of the obtained results in this work.

Through the second set of experiments, we aim at showcasing how the online and distributed sensing capabilities of the proposed framework can aid towards realistic evaluation of wireless protocols in terms of energy efficiency. In this context, we conduct two experiments that implement an everyday life scenario of simultaneous file uploading by multiple peers, under varying configurations and channel conditions. In parallel with the experiment execution, we monitor the power consumption of each node's NIC and thus we manage to get a high level perception of the impact that the varying conditions and the simultaneous operation of multiple nodes induce in the overall energy consumption. We present these two sets of experiments in the following section and organise them in two different groups, namely the low-level and the high-level ones.



(a) Enabled A-MPDU aggregation NIC Power Consumption

(b) Enabled A-MPDU aggregation Atom Node Power Consumption

(c) Disabled A-MPDU aggregation NIC Power Consumption

Figure 67: Power Consumption of AR9380 NIC and Atom node during transmission across varying Application-Layer Traffic

8.5.1 *Low level Experiments*

The experimental setup in this first set of experiments consists of just two communicating nodes that operate on the vacant Ch.36 of the 5 GHz band. In this setup, we adjust specific PHY-layer configurations at the transmitter node, while we constantly monitor the energy consumption of the individual wireless NICs at both the transmitter and receiver side. Based on off-line processing of the collected results, we isolate specific events, such as frame transmission or reception and average multiple of them, in order to characterize their instantaneous power consumption, under each specific PHY-layer configuration. In the following experiments, we use two different Atheros chipsets, namely the AR5424 and AR9380 that implement different versions of the IEEE 802.11 protocol. The AR5424 was released in 2005 as the first complete single-chip of Atheros that supported the IEEE 802.11 a/g protocol, while the AR9380 chipset is currently the state-of-the-art IEEE 802.11n compatible Atheros chipset, featuring 3 RF chains and supporting up to three transmit and receive spatial streams (SS).

Chipset	AR5424	AR9380		
Antennas	1x1	1x1	2x2	3x3
Mode	Power Consumption (Watts)			
Sleep	-	0.12		
Idle	1.47	0.49	0.56	0.69
Receive	1.52	0.62	0.74	0.85
Transmit	1.97	0.98	1.75	2.45

Table 10: Power consumption of AR5424 and AR9380 NICs across different operational modes

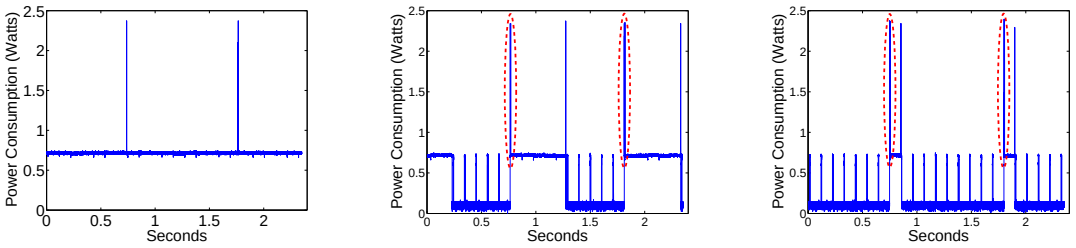
We start by characterising the power consumption profile of the two different wireless chipsets, across various operational modes, and present the collected results in Table 1. Based on the collected results, we observe that the AR9380 chipset consumes

much less power than the AR5424 chipset across all operational modes. This observation comes in contrast with the increased power consumption that multiple-input and multiple-output (MIMO) enabled chipsets are characterised by, due to the use of multiple radio chains and the complicated baseband processing that are required to enable for MIMO communication. However, the later manufactured MIMO-enabled chipsets feature advanced IC technology, which is the key feature that results in their significantly optimized energy efficiency in all operational states. Regarding the sleep mode power consumption of the AR5424, we remark that the generic IEEE 802.11 *Power Saving Mode (PSM)* mode is not supported by the open-source *Mad-WiFi* and *ath5k* drivers that control its operation and thus this value is not presented in Table 1. On the other hand, we were able to configure the PSM mode for the AR9380 NIC through the *ath9k* driver, which set the NIC in low-power state (0.12 W) by disabling most of its circuitry.

In the following experiments, we characterize the power consumption characteristics during frame transmission and reception under various settings, for both of the IEEE 802.11a/g and IEEE 802.11n compatible wireless NICs and present the obtained results in sections 5.1.1 and 5.1.2 accordingly. We also investigate the impact of the PSM mechanism on the consumption of wireless NICs and delay induced to the generated flows, through specifically designed experiments that are presented in section 5.1.3.

8.5.1.1 *Experimentation with IEEE 802.11a/g NICs*

In this experiment, we start by characterizing the instantaneous power consumption of the IEEE 802.11a/g compatible Atheros AR5424 chipset, during frame transmission and reception events. Frames are transmitted under fixed TX_R values between the 6 Mbps and 54 Mbps that are supported by the IEEE 802.11a/g standards. We also fix the Transmission Power (TX_P) of the transmitter node at the maximum value of 18 dBm. Based on the collected results, we observe that the consumed power, when transmitting or receiving frames under different TX_R, does not vary significantly. As duration of frame transmission is monotonically related to the configured TX_R, it is important to quantify energy efficiency in terms of energy consumption per transmitted bit of information (E_B). We calculate E_B, expressed in Joules/bit, as the division of the power consumption values collected for each different TX_R by the exact TX_R value expressed in bits/sec. Although E_B is a useful metric, it does not include the amount of energy that is spent for retransmission of undelivered frames, as it does not consider the frame delivery rate at the receiver, which significantly differentiates between different TX_R in realistic environments. In Fig. 65a, we plot the obtained E_B across the available IEEE 802.11a/g compatible TX_R configurations, considering both the transmission and re-



(a) PSM OFF (b) PSM ON - Timeout Period 500 ms (c) PSM ON - Timeout Period 100 ms

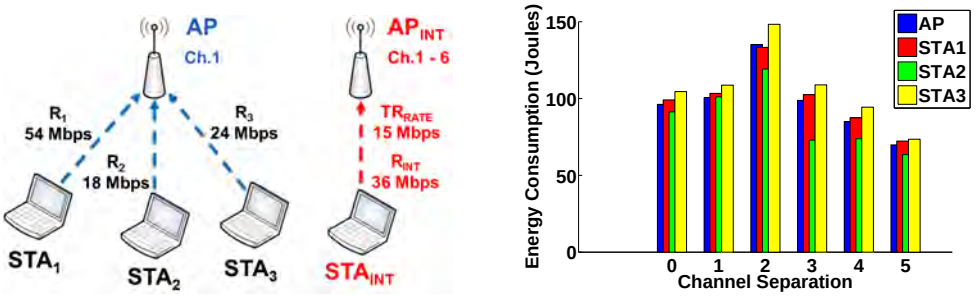
Figure 68: Instantaneous Power Consumption of AR9380 NIC and impact of PSM

ception events. We notice that higher TX_R settings always result in lower E_B, which is mainly due to the decreased duration of the transmission or reception duration.

Next, we fix the TX_R value and characterize the power consumption of frame transmissions under varying TX_P settings, by configuring the TX_P of the transmitter node at the 8 available settings between the values of 18dBm and 0 dBm. Fig. 65b presents the instantaneous power consumption across the 8 available TX_P configurations and under the minimum (6Mbps) and maximum (54 Mbps) available TX_R settings. We clearly notice that power consumption decreases monotonically with the decrease of the TX_P and that the decrease rate varies across different TX_R configurations. Moreover, we observe that the maximum achievable energy saving of 14% can be obtained in the case that the TX_P is reduced from the default value of 18 dBm to the minimum of odBm, in the case that TX_R is set to 6 Mbps.

8.5.1.2 *Experimentation with IEEE 802.11n NICs*

In this second experiment, we investigate the impact of low-level configurations on the energy consumption of the IEEE 802.11n /MIMO compatible Atheros AR9380 chipset. AR9380 can be configured in 3 different SS configurations, namely SISO, MIMO2 and MIMO3, where one, two, or three RF-chains are used accordingly. Moreover, each different spatial stream configuration offers 8 different modulation and coding schemes (MCS), resulting in up to 24 different MCS settings for the MIMO3 case. In this experiment, we transmit frames, under fixed MCS indexes among the ones offered in each SS configuration and calculate the instantaneous power consumption in each setting. More specifically, we use channel bandwidth of 20 MHz and guard interval of 800 ns, resulting in TX_R settings that range from 6 Mbps in the MCS0 case to 195Mbps in the MCS23 case.



(a) Experimental Topology

(b) Energy Consumption/ NIC on each channel

Figure 69: 1st high-level experiment using 802.11a/g compatible NICs

In order to calculate E_B for the various MCS settings, we follow the procedure described in the previous section. In Fig. 66a, we plot the obtained E_B across the 23 available TX_R configurations, in the case that all RF-chains are constantly enabled, even in cases that the configured SS setting does not use the excess antennas. Based on the collected results, we notice that MCS configurations significantly impact power consumption, as imposed by the calculated E_B, which ranges from 376.6 nJ/bit (MCS0) to 12.2 nJ/bit (MCS23). This finding suggests that there exists a huge potential for minimisation of energy expenditure (up to 97%), through proper adaptation of MCS configurations. Figures 66b and 66c present E_B measurements for transmission and reception accordingly, in the case that only the required number of RF-chains are enabled for each configured SS setting. We notice that proper activation of the required number of RF-chains (SISO, MIMO2) can significantly increase energy savings up to 60% for transmission (27% for reception), as for the MCS0 case, where E_B reduces to 153.2 nJ/bit (95.38 nJ/bit). More specifically, we also observe that MCS indexes within the same SS configuration do not remarkably impact power consumption, while indexes of different SS settings result in highly diverse E_B values, due to the power consumption values that significantly vary between different antenna settings (especially during transmission, as presented in Table 1).

The last part of this experiment has been designed to assess the impact of Application-layer Traffic Rate (TR_R) on the consumption of wireless NICs. In this experiment, we fix the MCS index to 23, resulting in the PHY-layer TX_R value of 195 Mbps. Under this fixed configuration, we run an experiment that varies the TR_R at the transmitter node in 6 steps, among the values of 10, 20, 50, 100, 200, 300 Mbps. The whole experiment runs for 60 seconds and approximately 4×10^6 voltage samples are collected.

Fig. 67a illustrates the power consumption of the NIC under the various configured TR_R values. We clearly observe that increment of Application-layer Traffic Rate results in increased power consumption, where the highest monitored increase of 0.13W, is

observed between the 10 and 300Mbps TR_R values. We also notice that this observation holds even when TR_R reaches the 300 Mbps value and increases above the capacity limit. However, it is unclear whether the increased energy expenditure is solely related to the increased amount of bytes or also related to the number of frames delivered to the driver. The work in [**?**], which considered energy consumption on the total node level, revealed and quantified that a substantial proportion of energy is consumed during the packet processing through the Operational System (OS) protocol stack. Moreover, this work suggests that the monitored energy expenditure on the total node level, is primarily associated to the frame processing itself, rather than to the amount of bytes handled. We managed to verify the findings of the work in [**?**], by measuring the power consumption of an Atom-based node equipped with the AR9380 NIC and by configuring the same TR_R settings. As illustrated in 67b, increasing application layer Traffic Rate values result in increased power consumption, where the highest monitored increase of approximately 3.5W, is observed between the 10 and 300Mbps TR_R values.

Furthermore, in order to assess the impact of PHY-layer A-MPDU frame aggregation on power consumption, we disable frame aggregation and repeat the same experiment. As demonstrated in Fig. 67c, A-MPDU aggregation slightly increases the resulting power consumption, as monitored for the lowest TR_R value. However, the monitored trend of increasing power consumption across increasing TR_R values, is not identified in this case. These observations are rather pioneering and yield interesting insights regarding the impact of traffic load, number of frames delivered to the driver and A-MPDU aggregation on energy consumed by the NIC.

8.5.1.3 *Experimentation with IEEE 802.11 PSM*

Through this experiment, we aim at quantifying the potential energy savings of the IEEE 802.11 PSM that is designed to set the wireless NICs of stations (STAs) in a low-power state during periods of inactivity and periodically activate them to fetch cached data from the access point (AP). PSM is controlled through *Beacon* frames that are transmitted by the AP at the constant *Beacon Interval* and include the *Delivery Traffic Indication Message* (DTIM), which is used to notify sleeping STAs about incoming data cached at the AP that are waiting to be delivered. Once the sleeping STA is being informed through the DTIM that frames are waiting to be delivered, it returns to active mode and transmits the *PS-Poll* frame to inform the AP that it is ready to receive the cached data. The DTIM may be reported every n *Beacon Intervals* and in this case the STA only needs to awaken at the DTIM interval to check for cached data. However, as shown in [143], most commercial cards are configured to wake up and receive all

transmitted *Beacon* frames. The last determines that the maximum period a WLAN NIC can remain in sleep mode is indicated by the *Beacon Interval*.

In this experiment, we use the AR9380 chipset, which supports the PSM mechanism and configure the default *Beacon Interval* of 100 ms. We use the ping application to transmit ICMP [144] packets every 1 second, from the AP to the STA and measure the delay induced during NIC sleep intervals. In parallel with the described experiment, we monitor the power consumption of the STA's NIC. First, we disable the PSM mechanism and as depicted in Fig. 68a, we observe that the STA's NIC consumes between 0.69 W and 0.74 W in idle and receive mode accordingly, while it consumes 2.45 W to transmit the *ICMP Echo Reply* packets, resulting in the total energy consumption of 1.6923 J within the 2.35 seconds of observation.

In the second phase, we activate the PSM mode, so that the STA's NIC constantly operates in sleep mode and is activated to receive *Beacon* frames every 100 ms and get informed about cached data. In Fig. 68b, we present the instantaneous power consumption of the STA's NIC and highlight in the red ellipses the instances during which the NIC transmits the *PS-Poll* frame and subsequently receives the *ICMP Echo* and transmits the *ICMP Echo Reply* packets. We also observe that the NIC does not directly sleep, but remains in idle mode for 500 ms, which is the default *Timeout Interval*. In this phase, the NIC consumes approximately 1.0613 J, resulting in an overall reduction of 37%, in comparison with the first phase. In Fig. 68c, we plot the consumption during the third phase, where we set the *Timeout Interval* equal to 100 ms and measure that the NIC consumes just 0.4635 J (73% reduction) during the observation interval. Regarding the ping application delay, we remark that it equals 1.3 ms in the first phase, while in both of the next two phases it approximates the 150 ms value, as the configuration of a shorter *Timeout Interval* does not impact the frequency of *ICMP Echo Reply* packet transmissions. Based on the obtained results, we remark that significant energy savings can be attained, through proper scheduling of sleep intervals, while maintaining delay within acceptable limits.

Having executed several low-level experiments, we managed to identify that significant amount of energy can be saved, through proper configuration of MAC protocol parameters. However, further investigation is required in order to arrive at safe conclusions and we aim at examining these initial observations in detail as part of our future work. Nevertheless, the extended set of presented low-level experiments have clearly demonstrated the advanced energy monitoring capabilities of the NITOS EMF framework.

8.5.2 *High level Experiments*

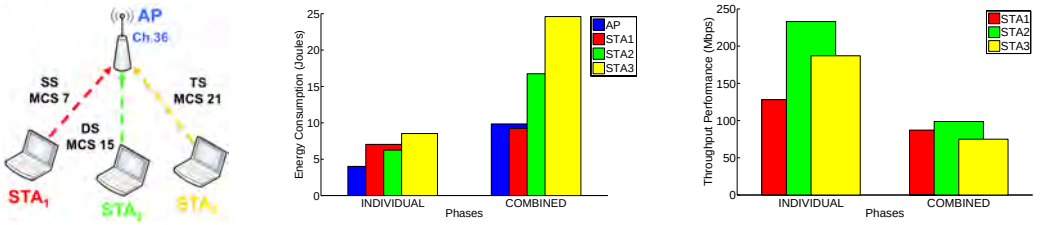
The experimental scenario in the following two experiments includes 3 STAs that are associated with the same AP and are simultaneously uploading a file of fixed size. The same experiment is repeated under varying channel conditions and different PHY-layer configurations for each wireless NIC, while energy consumption is being constantly monitored. In the first experiment, we use only IEEE 802.11a/g compatible NICs, while in the second one IEEE 802.11n compatible NICs are employed.

8.5.2.1 *First Experiment based on IEEE 802.11a/g*

In this experiment, the 3 STAs are simultaneously uploading a file with size of 25MBs, while being associated with the AP that is operating on Ch.1 of the 2.4 GHz band. Another pair of collocated nodes is generating interference, with the station node (STA_{INT}) transmitting on uplink at the Application-layer traffic load of 15Mbps. Fig. 69a illustrates the experimental topology along with the PHY-layer rate settings of each specific NIC. The experiment is repeated 6 times, where in each different run we configure the AP_{INT} to operate on a different channel between Ch.1 and Ch.6 of the 2.4 GHz band. The experimental setup is inspired by our previous work on Dynamic Frequency Selection Algorithms [111].

During each different run, we monitor the energy consumption of each NIC and plot the collected results in Fig. 69b. While AP_{INT} moves from Ch.1 to Ch.2 and subsequently to Ch.3, we notice that the total energy consumption of all NICs is increased. Due to adjacent channel interference, transmissions of STA_{INT} are not always detected by the three STAs, which results in frame collisions and subsequent frame retransmissions. The overall effect is that the file transmission durations are increased for each individual node and thus impact the overall energy expenditure. As AP_{INT} moves from Ch.3 to Ch.5, we notice that the energy consumption tends to decrease for all NICs, resulting in the lowest monitored values in the case that the interfering link operates on Ch.6, as it no more interferes with the 3 STAs.

A particular observation in all cases is related to the energy performance of STA2. While STA2 uses the lowest PHY-layer rate of 18 Mbps, compared to the rates of STA1 (54Mbps) and STA3 (24Mbps), it manages to result in the lowest energy expenditure in all cases. This comes in contrast with the higher E_B values that the lower PHY-layer rates correspond to. However, this uniquely identified performance results due to the fact that STA2 completes the file uploading sooner that the rest nodes, in all cases.



(a) Experimental Topology

(b) Energy Consumption/ NIC in each phase

(c) Throughput/ NIC in each phase

Figure 70: 2nd high-level experiment using 802.11n compatible NICs

Based on the experimental log files, we observe that STA2 always achieves the highest throughput. The increased throughput performance of STA2 can only be associated with the "Capture Effect" phenomenon [145], due to which certain topology configurations result in unfair throughput distribution for specific links.

8.5.2.2 *Second Experiment based on IEEE 802.11n*

In this second experiment, the 3 STAs follow the same file uploading process, while the AP is operating on Ch. 36 of the 5 GHz band, using a channel bandwidth of 40 MHz. However, in this scenario MIMO enabled NICs are used and the size of the file to be uploaded is 1 GB. We statically fix the single (SS) stream configuration for STA1, while STA2 and STA3 are configured with the double (DS) and triple (TS) stream setting accordingly. We also enable the default MCS adaptation mechanism of the driver, which results in the assignments of MCS7 for STA1, MCS15 for STA2 and MCS21 for STA3, as presented in Fig. 70a. The experiment is executed in two phases, where in the first phase each node completes the file uploading, using the medium in an *individual* way, as the rest two STAs are disabled. In the second phase, we configure the 3 STAs to transmit simultaneously and monitor energy performance, during the execution of the *combined* experiment.

Fig. 70b depicts the total energy consumed by each individual wireless NIC, as monitored during both phases of the experiment. In the first phase, we notice that the AP results in the lowest energy expenditure, which comes from the fact that it operates in reception mode, in which case the energy consumption is significantly less compared with the consumption during transmission. We also notice that STA3 using the TS mode consumes the highest amount of energy (8.53J), while STA1(7.04J) and STA2 (6.25J) follow accordingly. In this case, the energy performance of all nodes is directly related with the E_B that each different MCS index is characterized by, which values are 7.705nJ/bit, 7.535nJ/bit and 6.66nJ/bit for MCS21, MCS7 and MCS15 accordingly.

In the second phase, we observe that average energy consumption is significantly increased for all nodes, resulting due to the decrease in channel access opportunities and the corresponding increase of idle listening periods. More specifically, STA3 again consumes the highest amount of energy (24.61J), while STA2(16.75J) and STA1(9.22J) follow accordingly. In this case, energy performance cannot be associated with the E_B parameter, as it does not consider the amount of time spent during idle listening periods. Based on the throughput performance evaluation, which is illustrated in Fig. 70c, we notice that in the combined experiment all nodes achieve nearly equal performance, which results in approximately equal time spent for the completion of the file uploading process. In this case, NICs that enable higher number of RF-chains and result in higher instantaneous power consumption, eventually induce higher total energy consumption within the same experiment duration.

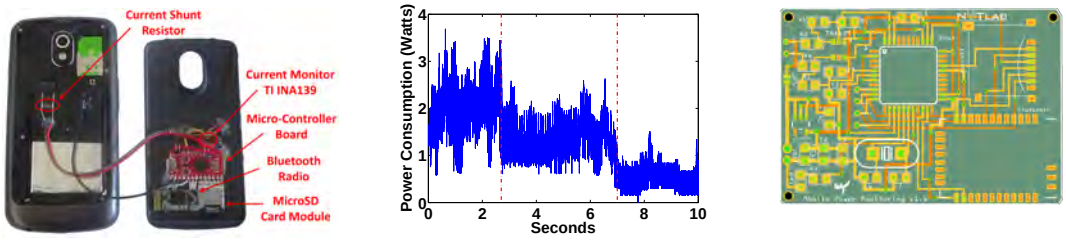
As it is clearly demonstrated through the two high level experiments, important factors, such as topology and interference conditions and interaction among simultaneously transmitting nodes, can greatly impact energy expenditure. Due to the inherent inability of power consumption models to accurately analyze energy consumption in such complex scenarios, online energy consumption monitoring needs to be applied, in order to arrive at concrete conclusions.

8.6 MONITORING OF MOBILE DEVICES

8.6.1 *NITOS Mobile Monitoring Solution*

NITOS EMF is designed to monitor the consumption of wireless testbed infrastructure at the level of the wireless NIC, as well as at the total node level. The developed solutions consist an ideal environment for conducting research related to the development of energy-efficient protocols and architectures. However, the main target of energy-aware protocols is to reduce the consumption of mobile devices, such as smartphones, tablets and laptops, which operate on batteries of limited capacity and are characterised by restricted duration of operation. Currently, NITOS EMF is able to monitor the consumption of mobile devices, however execution of realistic remote and mobile scenarios cannot be supported, due to the framework's mobility limitations.

In an effort to provide for energy efficiency evaluation of realistic experiments that include mobility scenarios as well, we designed the NITOS Mobile Monitoring Solution (MMS) and integrated it with mobile phones. The NITOS MMS follows a similar to the NITOS ACM design, as it is also based on the INA139 current-shunt monitor, while



(a) Prototype Development (b) Indicative Power Consumption plot (c) PCB

Figure 71: NITOS MMS

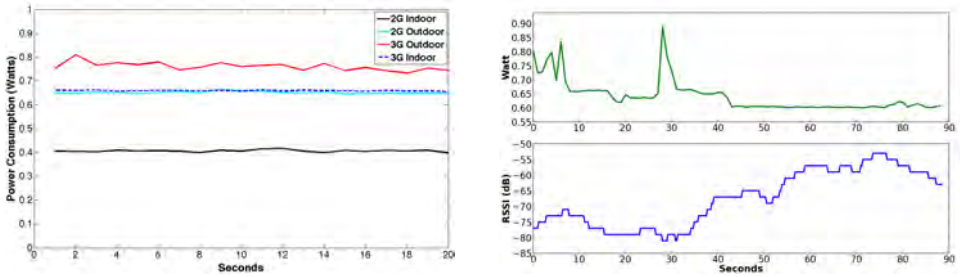
a Micro SD card module is also used for local storing of measurements. However, the Arduino Mega board has been replaced by the compact Arduino-like Pro Micro board [146] that operates at 3.3 V and features the ATmega32u4 [147] running at 8 MHz, while the Bluetooth interface RN-42-N [148] is used instead of the Ethernet interface for wireless remote control and transferring of measurements. Moreover, the mobile solution does not require external power supply, as it is properly powered by the mobile phone's battery, without affecting the power consumption measurement procedure. Fig. 71a presents the fully operational prototype solution and its various components integrated with a Samsung Galaxy Nexus mobile phone.

The maximum sampling rate of the Pro Micro has been increased from the default value of 4.33 KHz to 17 KHz through the application of techniques similar to the ones described in section 4.2. The low power consumption profile of the various selected components result in a total consumption of 0.043 W and 0.092 W, in idle and monitoring mode accordingly. Considering an experimental scenario that lasts for one hour, we estimate that the mobile solution will require approximately 30 mAH for power monitoring and additionally 8.2 mAH to transfer the 67.5 MBs generated file through the Bluetooth interface. As a result, the proposed measurement approach does not limit the operation of recent smartphone models, which feature batteries that exceed the capacity of 2000 mAH. Fig. 71b presents an indicative experiment, in which we monitor the power consumption of the Samsung Galaxy Nexus mobile phone. We highlight three phases, where during the first phase the phone's display is activated, then in the second phase and approximately at 2.8 s we power it off and observe that it takes approximately 4.2 s till the device falls in sleep mode in the third phase.

Towards the development of an integrated solution that will combine the various components, we started by designing and a Printed Circuit Board (PCB) (Fig. 71c). The developed device is able to fit in most mobile phone cases with minor modifications and has a total cost of the device is less than 35.

(a) Integration with Galaxy Nexus. (b) NITOS MMS. (c) NITOS MMS app.

Figure 72: NITOS MMS Hardware and Software Components.



(a) Average Power Consumption during 2G/3G calls in Outdoor/Indoor Locations. (b) RSSI and Corresponding Power Consumption.

Figure 73: Experimental Results.

The developed platform has been evaluated in comparison with the high-end NI-6210 data acquisition (DAQ) module [**?**] and proved of providing measurements of similar accuracy in the range under consideration. Moreover, as the proposed measurement procedure is rather generic, it can be directly applied to most of existing smartphones, since it only requires a minor modification of the battery. As Fig. 72a shows, our solution is properly powered by the phone's battery and thus does not require external power supply.

8.6.2 *NITOS Android Application*

NITOS MMS application designed and deployed for Android devices, is able to monitor all the information that the Android API exposes, concerning the phone's broadband interface. Each measurement set consists of several variables, such as RSSI, Cell ID, Network Type, Network Operator ID, timestamp and is cached on a local sqlite3 database. To ensure accurate monitoring, the application halts all the unnecessary processes and turns off the non-required interfaces, before triggering the monitoring procedure for both the MMS and mobile phone. Furthermore the application, which is illustrated in Fig. 72c, exploits any available network connection to off-load the collected data to NITOS [80] server for further processing.

8.6.3 *Experimental Evaluation*

The prototype device has been integrated with a Samsung Galaxy Nexus smartphone, as illustrated in Fig. 72a. In this section, we evaluate the developed device under a set of indicative experiments. In the first experiment, we start by characterising the instantaneous power consumption of the smartphone, during a phone call using the WCDMA 3G interface, while in the second phase we employ the GSM interface. In Fig. 73a, we plot the average power consumption, as measured in each different setup and observe higher power consumption for both the 3G and GSM networks in the outdoor environment.

In the second experiment, we showcase the impact of RSSI fluctuations on the energy consumption of the phone's cellular module. Exploiting the NITOS MMS application, we collect RSSI measurements alongside with the energy data logging, during a regular phone call. As demonstrated in Fig. 73b, low RSSI values correspond to higher energy consumption, as it is depicted in the first 40 seconds where the signal ranges from -83 to -68dB.

As part of our future work, we seek to integrate the NITOS MMS with mobile phones of volunteers, such as lab members and University students, in order to collect measurements corresponding to the energy that is consumed during long-term execution of everyday life scenarios.

8.7 CONCLUSIONS

In this work, we introduced the novel NITOS EMF framework that is able to characterise the consumption of wireless testbed infrastructure in an online way. The proposed framework is built on a distributed network of low-cost, but highly accurate devices and is fully integrated with the large-scale wireless NITOS testbed. Through extensive experiments, we demonstrated the advanced platform capabilities that can aid towards energy performance assessment of realistic testbed experiments. Finally, we presented the mobile version of the developed card, which is able to characterise power consumption of mobile devices under real life scenarios.

9

EXPERIMENTAL EVALUATION AND COMPARATIVE STUDY ON ENERGY EFFICIENCY OF IEEE 802.11

9.1 INTRODUCTION

IEEE 802.11 is currently considered as the default solution for implementing wireless local area network communications. The wide adoption of this standard by vendors of wireless devices offers high interoperability, which in combination with the provided ease of use and low deployment cost have resulted in its unprecedented market and everyday life penetration. While the base version of the standard was released in 1997, subsequent amendments have been proposed throughout the years, such as the widely adopted 802.11b, 802.11a and 802.11g. In 2007, the current standard IEEE 802.11-2007 [1] was released and merged several amendments with the base version.

In an effort to improve throughput performance of the base standard, the IEEE 802.11 standard working group started in 2003 to develop the IEEE 802.11n high-throughput (HT) extension of the base standard that was finally published in 2009. The most important improvement of the 802.11n on the Physical layer (PHY) is the ability to combine multiple antenna elements to achieve higher PHY bit rates and increased link reliability. In order to increase medium utilisation and exploit from the increased PHY bit rates, efficient Medium Access Control layer (MAC) frame aggregation mechanisms [149] are also supported. Several recent studies [150, 151] in the field of wireless networking have experimentally verified the improved channel efficiency and throughput performance that the 802.11n is able to deliver in comparison with 802.11a/g systems.

On the other hand, the recent penetration of 802.11n compatible chipsets in "smart" mobile devices has raised concerns regarding the energy efficient operation of the HT protocol. As the dramatically increased PHY bit rates, require the activation of multiple RF chains and complex baseband processing as well, 802.11n compatible chipsets induce significantly higher power consumption that grows with the number of active RF chains. Our experimental results that match other relevant studies [130, 152] as well, show that a modern 802.11n 3x3 MIMO compatible chipset is able to draw up to 2.45 W, in the case that all the various chipset components are constantly activated and the

highest PHY bit rate performance is achieved. Especially in the case of smartphone platforms, the energy greedy profile of the supported state-of-the-art wireless technologies may induce up to 50% of the total platform power consumption [6], under typical use case scenarios. As the focus of researchers is usually on network performance, only a few works have presented detailed experimental results that characterise the energy consumption of 802.11n chipsets [130, 152, 143]. The work in [130] is restricted in characterising the power consumption profile of commercial 802.11n chipsets, while the rest two studies [152, 143] experimentally investigate the impact of standard compliant power saving mechanisms on the operation of 802.11n protocol.

In this work, we take a step further than relevant studies and characterise the energy efficiency of 802.11 compliant protocols, in comparison with the achievable network performance they are able to offer. Our study is not restricted in evaluating the performance of 802.11n chipsets, but presents detailed experimentally obtained measurements that compare the performance of the legacy 802.11a/g standard with the latest 802.11n version. The impact of various MAC layer enhancements, both vendor specific and standard compliant ones, is also considered in the performance evaluation of both protocols. The obtained results are collected in realistic scenarios and under a wide range of settings, considering varying application-layer traffic loads, frame payload lengths and different network topologies that offer varying channel conditions as well. Finally the impact of the default 802.11 Power Saving Mechanism (PSM) on the performance of both protocols during periods of activity has also been investigated.

Accurate energy efficiency evaluation under real world scale and settings, is a rather complex task that requires the application of detailed evaluation methodologies, in combination with advanced power monitoring platforms. The followed power measurement methodology is based on custom-built hardware that was developed in our previous work [101] and enables online energy consumption evaluation at both the Network Interface Card (NIC) and the total wireless node levels. In-depth analysis of the extensive list of obtained measurements aided in identifying factors that affect energy consumption on commodity 802.11 hardware. Our detailed findings that can act as guidelines towards designing energy efficient wireless protocols, are summarised as follows:

- Activation of additional RF chains that enable MIMO communications results in remarkably increased power consumption (up to 2.5x) at the NIC level. However, our experiments have shown that the resulting increased PHY bit rates of 802.11n are able to increase energy efficiency at the NIC level by 33% during transmission (63% in reception), in comparison with the rates of 802.11a/g. Moreover, we

observed that proper activation of the required number of RF-chains for each specific rate configuration can aid towards saving energy.

- Application of MAC-layer aggregation mechanisms is able to deliver substantially increased throughput, while also resulting in considerable energy savings. Power consumption experiments that consider consumption at the total node level have shown that the aggregation assisted 802.11n can improve energy efficiency by more than 80%, in comparison with the performance of the 802.11a/g standard.

- While transmitting MAC frames of low payload length and under high-SNR conditions, 802.11n increased energy efficiency at the node level by 90%, in comparison with 802.11a/g. This observation is related to the fact that the supported by 802.11n aggregation mechanisms enable delivery of high throughput performance (>100 Mbps), even when transmitting frames of 300 bytes payload.

- Considering low-SNR conditions, the Spatial Diversity mode of 802.11n offers increased MAC-layer Frame Delivery Rate (FDR) and throughput improvement by a factor of 4.6x, as observed in our experiments. In addition, we remark that the monitored throughput improvement did not induce significant energy costs, as the energy efficiency at the NIC level also increased by 58%.

- Experimentation with the default 802.11 PSM mechanism has shown that 802.11n is able to provide significant energy savings (> 75%), across varying traffic loads and without sacrificing application-layer throughput or jitter performance.

This paper is organized as follows. In Section 2 we present the evolution of the base 802.11 standard over the last years. Section 3 details the experimental setup and the followed power measurement methodology that is used in our experimental evaluation. In Section 4, we characterise the power consumption profiles of the two wireless chipsets that are used in our experiments, while in Section 5 we present extensive experiments that compare the performance of the 802.11a/g and 802.11n protocols in terms of network performance and energy consumption. Finally, in Section 6 we point out the conclusions reached through this work.

9.2 EVOLUTION OF IEEE 802.11

The aforementioned versions of IEEE 802.11 use different PHY layer specifications, but all follow the MAC architecture of the base protocol. The mandatory access scheme that has been specified by the legacy IEEE 802.11 standard is implemented through the

distributed coordination function (DCF), which is based on the carrier sense multiple access with collision avoidance (CSMA/CA) mechanism. The large PHY and MAC layer overheads that are associated with the DCF process, result in a reduction of more than 50% of the nominal link capacity, which effect is more pronounced for higher PHY bit rates, as shown in [153]. The work in [154] has analyzed the throughput and delay limits of the IEEE 802.11 standard and has shown that for infinitely high PHY bit rate and a frame payload size of 1024 bytes, the maximum achievable throughput is upper bounded to 50.2 Mbps. Such observations highlighted that MAC layer enhancements need to be applied, in order to reduce the impact of the PHY and MAC layer overheads of the base standard.

In an effort to improve throughput performance, vendors of wireless products started integrating innovative techniques into their products, as early as 2003. Such techniques include the "Atheros Fast Frames" (*FF*) [155], which improves 802.11a/b/g performance, by combining two MAC frames into the payload of a single aggregated frame. However, application of vendor-specific techniques has been reported to result in hardware incompatibilities, or at least degraded performance for standard compliant devices, as presented in [156].

Along the same direction, the IEEE 802.11 standard working group introduced the 802.11n extension that offers both PHY and MAC layer enhancements over legacy 802.11 systems. Through the combination of multiple antenna elements and complex MIMO processing, 802.11n is able to achieve higher PHY bit rates (in Spatial Multiplexing mode) and increased link reliability through the exploitation of multipath transmissions and antenna diversity (in Spatial Diversity mode) [157]. Another significant feature is the application of channel bonding, which increases the channel bandwidth from 20 MHz to 40 MHz and thus doubles the theoretical capacity limits. Moreover, the available Modulation and Coding Schemes (MCS) were extended through the introduction of the new coding rate of 5/6, as well as through the decrease of the OFDM guard interval from 0.8 μs to 0.4 μs. Finally, the number of OFDM data subcarriers was increased from 48 to 52, towards improving spectral efficiency. Application of the aforementioned enhancements is able to deliver the remarkably increased PHY bit rate of 600 Mbps (when using 4 antennas), resulting in performance improvement of more than 10x compared to legacy 802.11a/g systems.

In order to increase medium utilisation and exploit from the increased PHY bit rates, two different types of frame aggregation are provided, namely A-MSDU and A-MPDU aggregation. The former combines multiple higher layer packets into a single MAC layer frame with maximum size of 7935 bytes, while the latter combines multiple MAC layer frames to form an aggregated frame that cannot exceed the 65.536 bytes. In gen-

eral, A-MPDU aggregation outperforms A-MSDU, which technique results in considerably degraded performance under low quality channel conditions and high PHY bit rates, as it was shown in [158]. Both frame aggregation mechanisms are enhanced by a block acknowledgment mechanism, which further reduces protocol overhead.

9.3 MEASUREMENT SETUP

In this section, we present the measurement setup that is used in the detailed experimental evaluation that aims at deriving a comparative performance analysis between the 802.11a/g and 802.11n standards. We start by describing the exact experimental setup, which is based on commercial wireless NICs that are representative of the state-of-the-art of each standard. In addition, we detail the followed power measurement procedure and the underlying hardware that are used to characterise the energy consumption performance of the considered protocols. The experimental setup that is used as the basis of our evaluation, consists of a single communicating pair of nodes that both feature the specifications listed in Table 11.

Component	Type
Motherboard	Commell LE-575X
CPU	Intel Atom D525 (1.8 GHz)
RAM	Kingston HYPERX DDR3 - 4GBs
Hard Drive	Samsung SSD - 64 GBs
Power Supply	60W - 12V
OS	Ubuntu 13.04
Wireless cards	Atheros 9380 / 5424
Wireless Drivers	madwifi-0.9.4 / backportsv3.12.1

Table 11: Node Specifications

Wireless communication is enabled through the Atheros AR5424 and AR9380 chipsets that implement the 802.11a/g and 802.11n protocols and are configured through the use of the *Mad-WiFi* [**?**] and *ath9k* [**?**] open source drivers accordingly. The wireless nodes are closely located in an indoor office environment at the University of Thessaly premises and are configured to operate in infrastructure mode, on the RF-isolated channel 36 of the 5 GHz band, in order to constantly guarantee un-interfered communication. We setup two different topologies, by keeping the AP node at the same physical location, while we move the STA between the two locations that are depicted in Fig. 74.

We use the Iperf [58] tool to generate traffic and collect network performance statistics. A typical experimental setup for experiments considering downlink transmissions, would be to run an Iperf client at the AP node, having also an Iperf server residing at the STA, receiving the traffic and collecting statistics. Moreover, we also exploit from the statistics that the applied Rate Adaptation algorithm of each wireless driver is able to export, in order to measure link reliability in terms of MAC-layer FDR, as calculated per each configured PHY-layer rate.

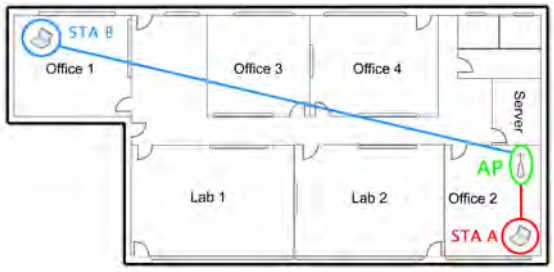


Figure 74: Experimental Topology

9.3.1 *Experimental Setup*

We first place the STA at location A to establish a line-of-sight high-SNR link (Signal to Noise Ratio (SNR) ~ 35 dB), by configuring the transmission power of both nodes at the maximum level of 20 dBm. Towards executing experiments under low-SNR (SNR ~ 15 dB) channel conditions, we move the STA to location B and reduce the transmission power of both nodes to the minimum available level of 0 dBm. The aforementioned SNR values correspond to the 802.11a/g link, while the 3x3 MIMO configuration of the 802.11n link provides approximately 5dB gain in each setup, by exploiting spatial diversity at the receiver through the Maximal-ratio Combining (MRC) technique [159].

Towards providing for a proper comparison setup between the two standards, we configure both transceivers to use the same channel bandwidth of 20 MHz and OFDM guard interval of 0.8 μs. Under this setup, we execute each discrete experiment in two phases, where in each phase either the 802.11a/g or the 802.11n protocols are configured through the use of the corresponding transceivers. As the AR9380 chipset is also able to operate in the 802.11a/g mode, we measure its performance under this configuration, across the various considered cases as well. Under these settings, the 802.11n compatible chipset supports the maximum PHY bit rate values of 65 Mbps, 130 Mbps and 195 Mbps for single, double and triple spatial stream configurations accordingly, while the 802.11a/g compatible chipsets support PHY bit rate values between 6 Mbps and 54 Mbps.

Chipset	AR5424	AR9380		
Antennas	1x1	1x1	2x2	3x3
Mode	Power Consumption (Watts)			
Sleep	-	0.12		
Idle	1.47	0.49	0.56	0.69
Receive	1.52	0.62	0.74	0.85
Transmit	1.97	0.98	1.75	2.45

Table 12: Power consumption of AR5424 and AR9380 NICs across different operational modes

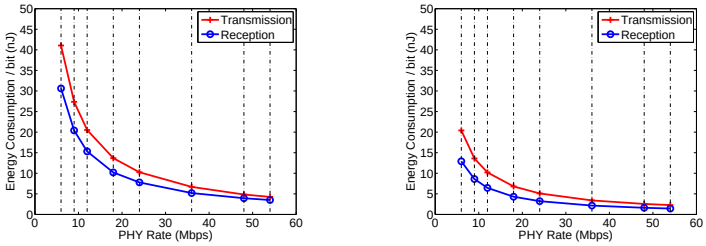
9.3.2 *Power Measurement Methodology*

In order to accurately measure the instantaneous power consumption, we follow a widely adopted power measurement procedure, which requires the placement of a high-precision, low impedance current-shunt resistor (R) of a known resistance value, in series with the power source and the power supply pin of the device to be measured. The exact measurement approach has been described in Chapter 7. In our study, we consider power consumption at both the total node level, as well as at the level of the wireless NIC. As a result, we decided to equip both nodes with appropriate current shunt resistors that have been placed in series with the power supply of the NIC and the Atom-based node accordingly. In order to ease the interception of the NIC power supply pins and refrain from modifying each different type of card, we decided to insert the current-shunt resistor on a PCI-e to mini PCI-e adapter card that is compatible with both wireless cards. The developed card supports the high sampling rate of 63 KHz and features up to three input channels, thus providing for online power consumption monitoring at both the NIC and the total node level, in a joint way.

9.4 POWER CONSUMPTION PROFILING

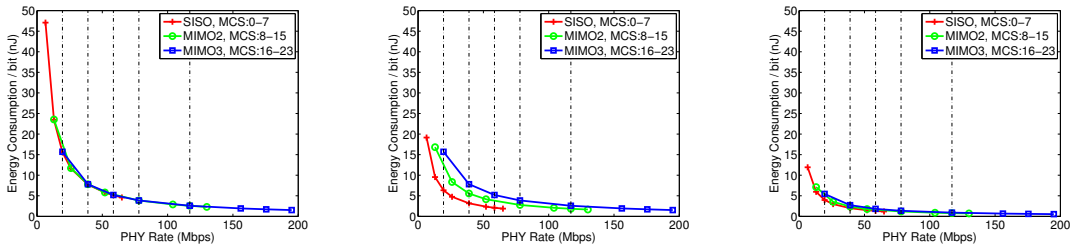
This initial set of experiments has been designed to clearly describe the power consumption profile of each chipset and set the basis for the realistic performance evaluation that follows in Section 5. Based upon the high-SNR setup, we characterize the instantaneous power consumption of the two NICs across various operational modes and present the obtained results in Table 12.

We clearly observe that the later manufactured AR9380 chipset is highly optimised for energy efficiency, as it consumes far less than 50% power than the AR5424 chipset, under all operational modes in the single antenna configuration. While considering MIMO operation, we notice that the activation of additional RF chains remarkably in-



(a) AR5424 (b) AR9380

Figure 75: Energy consumption/bit of 802.11a/g NICs across available PHY bit rate configurations



(a) Transmission - All RF chains constantly enabled (b) Transmission - Only required number of RF chains enabled (c) Reception - Only required number of RF chains enabled

Figure 76: Energy consumption/bit of 802.11n NIC across available PHY bit rate configurations

creases power consumption, as several hardware components need to be activated, in order to provide for the complex baseband processing that MIMO communications require. However, it is interesting to note that only in the case that 3x3 MIMO transmissions are executed by the AR9380, its power consumption increases above the consumption levels of the AR5424 chipset. Considering the power consumption in the sleep mode of operation, we remark that the *Mad-WiFi* driver does not support the activation of the *Power Saving Mode (PSM)* for the 802.11a/g compatible chipset. On the other hand, the PSM mode can be activated for the AR9380 NIC through the *ath9k* driver and set the card in a low-power state (82% less than in idle mode), by disabling most of the NIC's circuitry.

In order to assess the impact of varying PHY bit rates on NIC energy efficiency, we next proceed by characterising the energy consumption per transmitted bit of information (E_B), under the various PHY bit rate configurations that each protocol supports. We calculate E_B, expressed in Joules/bit, as the division of the resulting power consumption (Joules/sec) under each operational mode, by the specified PHY bit rate value expressed in bits/sec. In Figures 75a and 75b, we plot the obtained E_B across the available IEEE 802.11a/g compatible PHY bit rate configurations, for the AR5424 and the AR9380 chipsets accordingly. In the case that the AR9380 chipset is configured to operate in the IEEE 802.11a/g mode, we manually disable the excess RF chains and use the single mode antenna of operation, thus resulting in the significantly reduced E_B

values. Based on the obtained results, we remark that instantaneous power consumption of both chipsets does not significantly vary between different PHY bit rate settings. On the other hand, as plotted in Figures 75a and 75b , higher PHY bit rate settings always result in lower E_B, mainly due to the decreased duration of the transmission or reception operations.

We next proceed to the characterization of the power consumption profile of the 802.11n compatible AR9380 chipset, which offers a wider range of available MCS configurations, as it features three RF-chains and supports up to 3x3 MIMO mode of operation. In Fig. 76a, we plot the obtained E_B across the 23 available MCS configurations, in the case that all RF-chains are constantly enabled. We clearly notice that MCS configurations significantly impact power consumption, as imposed by the calculated E_B, which ranges from 47.1 nJ/bit (MCS0) to 1.53 nJ/bit (MCS23). This finding indicates the huge potential of energy expenditure minimisation (up to 97%), through proper adaptation of MCS configurations, as employed in similar studies [129].

In the case that all RF-chains are continuously enabled and while operating in the SISO and MIMO2 modes, the excess antennas do not contribute to the PHY bit rate increase, but are only used to provide increased link reliability. However, under ideal channel conditions, the excess antennas are no longer required to improve link reliability and thus can be deactivated, towards reducing energy expenditure. Figures 76b and 76c present E_B measurements for transmission and reception accordingly, in the case that only the required number of RF-chains are enabled for each configured SS setting. We notice that proper activation of the required number of RF-chains (SISO, MIMO2) can significantly increase energy savings up to 60% for transmission (27% for reception), as for the MCS0 case, where E_B reduces to 19.15 nJ/bit (11.92 nJ/bit). Considering the 802.11n configuration of the AR9380 chipset, we observe that its instantaneous power consumption does not significantly vary between different MCS indexes within the same SS configuration. Based on this fact, we infer that the high diversity of E_B values that is plotted in Figures 76b and 76c, is mainly due to the increased power consumption that activation of additional RF-chains results in.

Based on direct comparison of the E_B values plotted in Figures 75b, 76b and 76c, we aim at quantifying the energy savings that 802.11n can offer in contrast to the earlier 802.11a/g protocol. Considering the power consumption performance of the AR9380 chipset at the highest configurable PHY rates of each protocol (54 Mbps for 802.11a/g and 195 Mbps for 802.11n), we observe that energy savings of up to 33% can be attained during transmission (63% during reception), as E_B reduces from 2.27 nJ/bit to 1.53 nJ/bit (1.44 nJ/bit to 0.54 nJ/bit in reception). Concluding, we note that the remarkably higher rates of 802.11n protocol can offer significant energy savings when

combined with proper adaptation of antenna modes, in comparison with the energy consumption performance of the earlier 802.11a/g protocol. However, the power consumption profile characterisation is only based on static PHY bit rate configurations and does not consider protocol overheads, MAC-layer parameters, such as aggregation, or link performance and their impact on energy efficiency. In the following section, we proceed by comparatively evaluating the performance of the two standards under realistic throughput experiments, considering various protocol parameters as well.

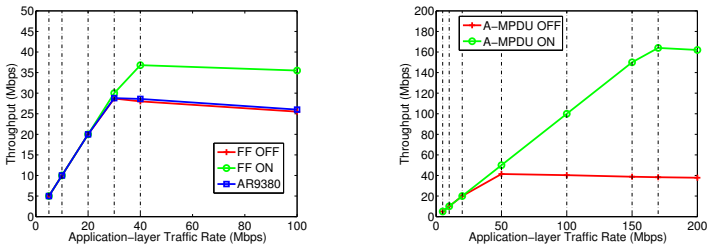
9.5 REALISTIC EXPERIMENTATION

In this section, we take a step further from characterising energy consumption under fixed modes of operation and conduct extensive realistic experiments to compare the performance of the two standards, by jointly considering application-layer performance and energy efficiency as well. We start by configuring the high-SNR setup to investigate the impact of varying application-layer traffic rate and frame payload length on the performance of each protocol. The obtained results are analyzed in Sections 5.1 and 5.2, for each scenario accordingly. In both scenarios, we manually configure the maximum available PHY bit rates of each standard and guarantee that these rates can be supported by the prevailing channel conditions, by constantly monitoring the achievable FDR and assuring that it never drops below 95% in all the conducted experiments.

Under these conditions, we start by measuring the throughput performance of each protocol without enabling any form of aggregation, while we next repeat the same experiments by explicitly enabling the *FF* and A-MPDU aggregation mechanisms. In each experiment, we also monitor the power consumption at both the NIC and the total node level, in order to assess the impact of the various configured settings on energy consumption. Next, in Section 5.2.2, we configure the low-SNR experimental setup to conduct series of experiments under varying frame payload size and investigate the impact on performance of both protocols, under low link quality conditions. Finally, in section 5.3, we assess the potential energy savings that the application of the PSM mechanism is able offer, under varying application-layer traffic rates.

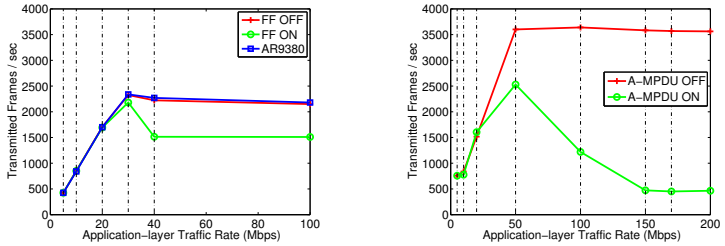
9.5.1 *Varying Application-Layer Traffic load*

We start by measuring the throughput performance under perfect channel conditions and across varying application layer traffic loads. The obtained results for the 802.11a/g and 802.11n protocols are illustrated in Figures 77a and 77b accordingly. In Fig. 77a, we observe that below channel saturation, throughput performance is similar



(a) AR5424 (b) AR9380

Figure 77: Throughput performance per NIC across varying Application-Layer Traffic load

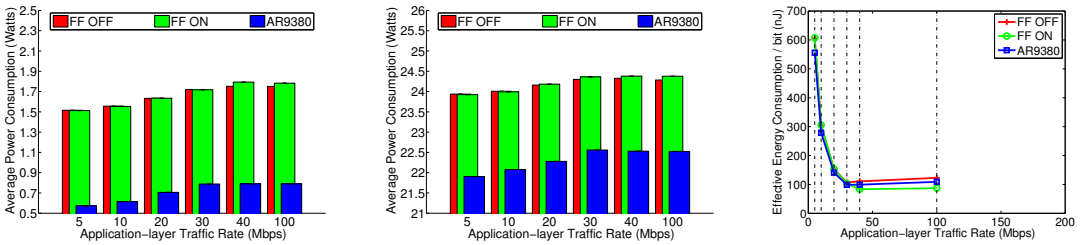


(a) AR5424 (b) AR9380

Figure 78: MAC-layer Frame Transmission rate per NIC across varying Application-Layer Traffic load

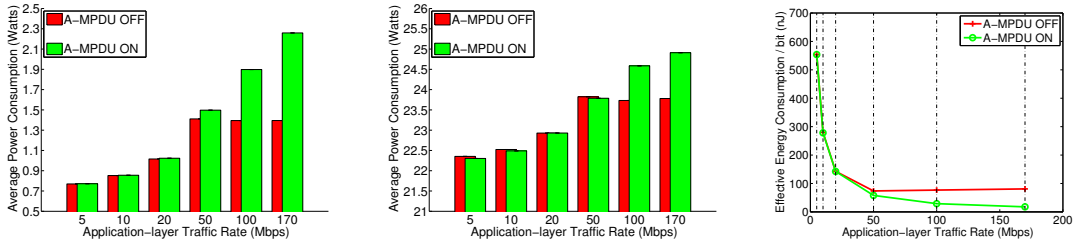
for both chipsets, while we also notice that the *FF* mechanism does not induce any impact. On the other hand and as soon as the load approaches 40 Mbps, we observe that the application of the *FF* mechanism offers approximately 31.4% increase in the maximum achievable throughput from 28 Mbps (*FF* disabled) to 36.8 Mbps (*FF* enabled). Based on detailed study of the *Mad-WiFi* driver code, we concluded that *FF* is only activated when the driver detects that the channel is approaching saturation through inspection of the transmission queue levels. We also verified our findings by monitoring the number of MAC-layer frames that are being transmitted in each time instant. In Fig. 78a, we plot the collected results and observe that in the 40 Mbps load case, the frame transmission rate decreases from 2150 to 1510 frames/sec (30% decrease).

Based on the 802.11n compatible setup, we repeat identical experiments and plot the collected results in Figures 77b and 78b accordingly. In the case that A-MPDU aggregation is disabled, channel reaches the saturation point as soon as traffic load equals 50 Mbps, while in the A-MPDU enabled case, saturation is only reached at the traffic load of 170 Mbps. Similar observations were made regarding the activation of A-MPDU aggregation, which is only activated when the channel approaches saturation (50 Mbps). Considering the 170 Mbps load case, we observe that A-MPDU aggregation increases throughput from 38.3 Mbps to 164 Mbps (4.3x increase) and decreases the MAC frame transmission rate from 3562 to 465 frames/sec. Our findings clearly verify that MAC layer improvements need to be applied, in order to exploit from the increased PHY bit rates that 802.11n offers.



(a) NIC (b) Total Node (c) Total Node Effective Eb

Figure 79: Energy efficiency characterisation of 802.11a/g setup across varying Application-Layer Traffic load



(a) NIC (b) Total Node (c) Total Node Effective Eb

Figure 80: Energy efficiency characterisation of 802.11n setup across varying Application-Layer Traffic load

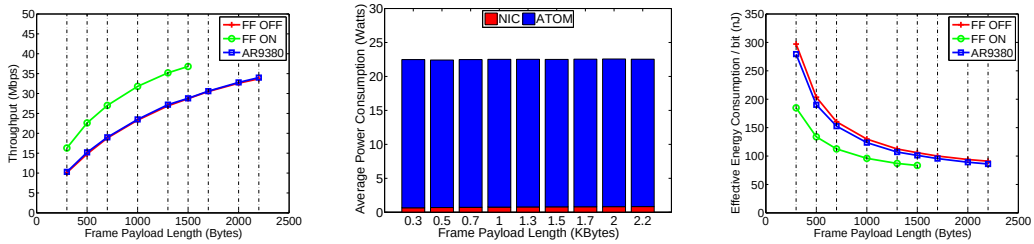
Having extensively evaluated the throughput performance improvement that the 802.11n protocol can offer across the various considered traffic loads, we next investigate how the monitored improvement is related with the resulting energy consumption. Figures 79a and 79b illustrate the average power consumption of both the 802.11a/g compatible chipsets and the total Atom node across the various configured traffic load values. As expected, average power consumption at both the NIC and total node level increases at higher traffic loads, due to the increased frequency of frame transmissions at the NIC level and the increased rate of frames that are being processed at the node level. In Fig. 79a, we observe that the NIC consumes between 1.55 W and 1.73 W, in both the *FF* enabled and disabled case, as the *FF* mechanism is not yet activated. As soon as the traffic load increases above 30 Mbps, *FF* is activated and average power consumption for the *FF* enabled case increases above the average monitored consumption for the *FF* disabled case, till it reaches the maximum value of 1.79 W. This observation comes due to the fact that in the *FF* enabled case, the NIC consumes more power on average as it operates in transmit mode for longer duration. Considering the power consumption of the total Atom node, we observe that the two different 802.11a/g based setups consume different amounts of power on average, due to the use different wireless chipsets and drivers. However, both setups witness an increase of approximately 0.5 W, as the traffic load increases from 5 Mbps to 30 Mbps. In the case of the *FF* enabled 802.11a/g setup, we observe that average power consumption at the total node

level is decreased between the 30 Mbps (22.56 W) and the 40 Mbps (22.53 W), in spite of the throughput performance increase. This observation is related with the activation of the *FF* aggregation mechanism, which efficiently reduces the rate of MAC-layer frames that are being processed by the driver. Our findings are summarised in the Effective E_B representation in Fig. 79c, which characterises the total node power consumption as a function of the resulting throughput and not as a function of the configured PHY bit rate. The obtained results clearly show that *FF* is able to reduce energy expenditure at the Atom-based node level, even up to 28% in the 40 Mbps case (119.5 nJ/bit FF OFF - 85.51 nJ/bit FF ON). Regarding the performance of the AR9380 equipped wireless node, we remark that in spite of its low power consumption profile, the resulting Effective E_B values are higher than the levels achieved by the application of the *FF* mechanism.

Similar results are obtained while evaluating the impact of A-MPDU aggregation on the power consumption of the AR9380 NIC and the Atom node, which are plotted in Fig. 80a and Fig. 80b accordingly. We clearly observe that A-MPDU aggregation results in significantly higher average power consumption, for traffic loads above 100 Mbps, both at the NIC as well as at the total node level, as a result of the achievable throughput gains. Comparing the consumption of the Atom node, as plotted in Figures 79b and 80b, we observe that under low traffic loads (< 50 Mbps) both protocols result is similar power consumption behaviour. This comes due to the fact the A-MPDU mechanism is not yet activated and as the high PHY rates of 802.11n are only able to reduce the average power consumption at the NIC level, the consumption of the total node is only minimally impacted. However, while considering traffic loads above 100 Mbps, we notice that the remarkably increased throughput performance that A-MPDU aggregation results in does not come at much higher energy costs. The Effective E_B representation in Fig. 80c, summarises the above results and shows that A-MPDU aggregation can increase energy efficiency up to 78%. Finally, direct comparison of E_B values at the corresponding saturation points of each standard shows that 802.11n offers more that 80% reduction of E_B compared with the AR9380 based 802.11a/g standard.

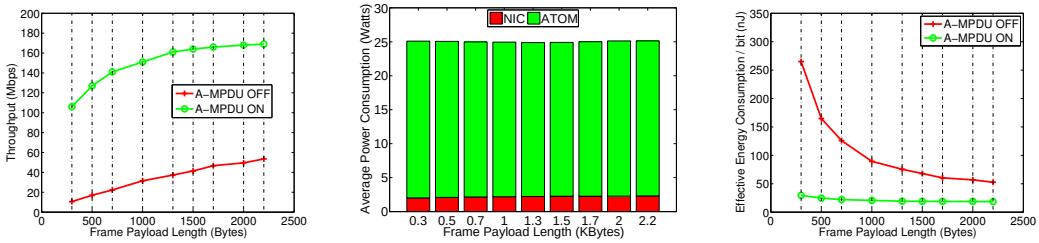
9.5.2 *Varying Frame Payload Length*

Extensive throughput experiments were also conducted under varying frame payload lengths, in order to investigate how varying Payload size affects throughput performance and energy expenditure. In order to enable delivery of frames longer than 1500 bytes to the MAC layer, we configured the wireless NIC's Maximum Transmission Unit (MTU) size to the maximum supported value of 2304 bytes. We have to mention



(a) Throughput (b) Average Power consumption (c) Total Node Effective Eb

Figure 81: Performance of 802.11a/g across varying Frame Payload Length values under high-SNR conditions



(a) Throughput (b) Average Power consumption (c) Total Node Effective Eb

Figure 82: Performance of 802.11n across varying Frame Payload Length values under high-SNR conditions

that in cases where the *FF* mechanism is enabled, aggregation of frames longer than 1700 bytes could not be handled by the driver, as the transmission duration exceeded the threshold of 4 ms that the 802.11 standard specifies as the maximum acceptable frame transmission duration. Having investigated the impact of varying frame payload length on performance under high SNR conditions, we next proceed by conducting identical experiments in the low-SNR experimental setup. The full list of obtained results are detailed in the corresponding sections that follow.

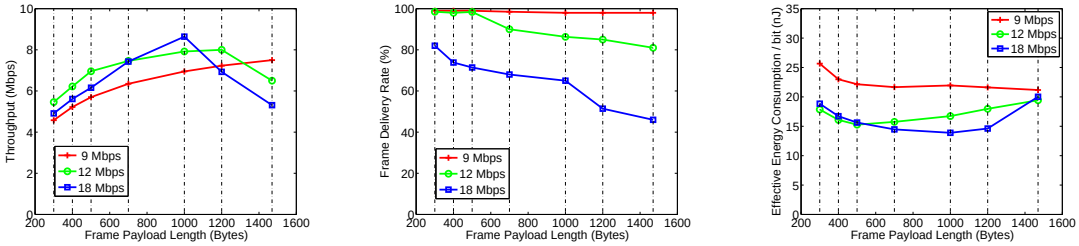
9.5.2.1 *High SNR conditions*

The throughput performance of the 802.11a/g and 802.11n protocols are illustrated in Figures 81a and 82a accordingly. We observe that under high-SNR conditions, increasing frame length values consistently result in improved throughput performance for both protocols. The throughput improvement between the lowest (300) and highest (2200) considered payload lengths varies between the factors of 3x and 5x for all the considered cases, except for the A-MPDU assisted 802.11n scenario, where the improvement is restricted in the order of 1.6x. This observation is related to the fact that A-MPDU aggregation enables 802.11n to deliver high throughput performance (>100

Mbps) even at the lowest payload length, as A-MPDU frame size is only limited by the maximum number of subframes (64) and maximum A-MPDU length (65.535 bytes).

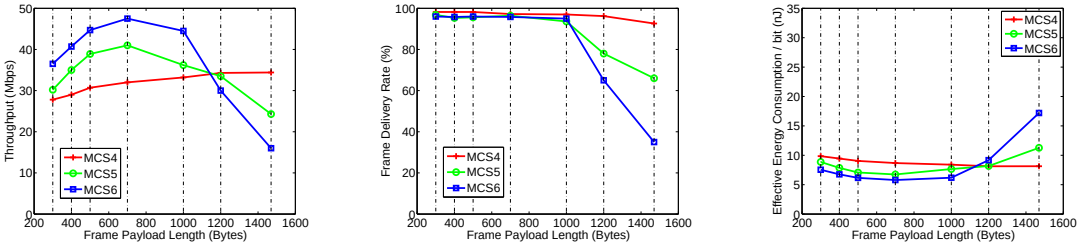
Figures 81b and 82b plot the power consumption of the AR9380 based setup, as it resulted during the operation of the 802.11a/g and A-MPDU-assisted 802.11n protocols accordingly. The full list of presented measurements have been collected using only the AR9380 based setup, in order to provide for direct comparison between the two protocols. Considering the minimum and maximum payload lengths for both protocols, we observe an increase of approximately 0.18 W in the consumption of the 802.11a/g configured NIC (0.3W 802.11n). The observed increase is directly related with the longer duration that the NIC remains in transmission mode, while achieving higher throughput. Regarding the consumption at the node level, in general we did not observe any significant consumption variation across varying payload lengths, which fact comes in contrast with the results obtained in the previous section, where even minor throughput improvement resulted in consumption increase at the total node level. In Fig. 82b, we even observe that the average power consumption decreases when the frame payload size increases from 300 bytes (25.1W) to 1300 bytes (24.88 W), Considering also that the NIC's average power consumption is also increased between these two cases, we remark that the consumption increase at the total node level approximates 0.5 W. As a result, we reach the conclusion that considerable amounts of energy are consumed while each frame crosses the protocol stack and verify the findings of the work in [**?**]. This observation in combination with the high throughput gains that payload increase results in, indicate that the use of longer frames is preferable in both terms of network performance and energy efficiency.

Our findings are summarised in the Effective E_B representation in Figures 81c and 82c. In comparison with the AR9380 based 802.11a/g setup, we remark that the A-MPDU assisted 802.11n is able to reduce the Effective E_B at the node level, from 279 nJ/bit to 29 nJ/bit (-90%) and also from 86 nJ/bit to 18 nJ/bit (-80%), when transmitting frames of 300 and 2200 bytes accordingly. Recent studies [74] of the packet size distributions in Internet traffic have shown that the most common packet lengths are of 576 bytes size, which fact highlights even more the energy savings that can be attained through the application of the 802.11n protocol. In addition, we remark that low frame lengths are usually preferable in the wireless domain, as they are able to provide increased FDR, especially when using complex modulation schemes that are susceptible to low-SNR conditions. This observation yields interesting insights and motivates further investigation regarding the performance of 802.11n across varying frame payload lengths and low-SNR conditions.



(a) Throughput (b) FDR (c) NIC Effective Eb

Figure 83: Performance of 802.11a/g across varying Frame Payload Length values under low-SNR conditions



(a) Throughput (b) FDR (c) NIC Effective Eb

Figure 84: Performance of 802.11n across varying Frame Payload Length values under low-SNR conditions

9.5.2.2 *Low SNR conditions*

Towards executing identical experiments under low link quality conditions, we establish the low-SNR experimental setup. In the following experiments, we measure the performance of 802.11a/g considering only the AR9380 setup and compare it against the A-MPDU assisted 802.11n case. Under this setup, the 802.11a/g configuration is able to sustain the PHY bit rate of 18 Mbps, while the 802.11n setup is able to use up to the MCS6 configuration in the Spatial Diversity mode. In addition, we measure the performance of less complex modulation schemes and more specifically the rates of 9 Mbps and 12 Mbps for the 802.11a/g configuration, while MCS4 and MCS5 are also configured for the 802.11n protocol. Characteristics of the various configured modulation schemes are listed in Table 13.

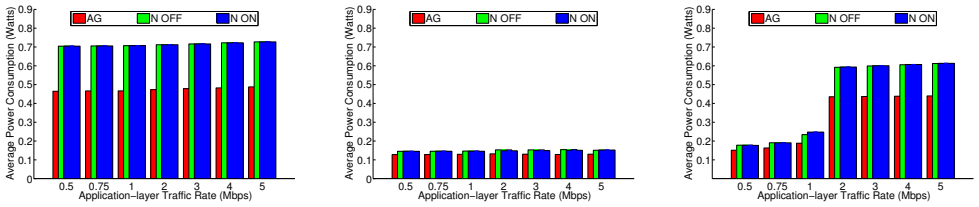
We start by measuring the throughput performance across varying frame payload lengths, between 300 and 1500 bytes, which is the default MTU size for the wireless chipsets under consideration. In Figures 83a and 84a, we illustrate the throughput performance that is achieved by each protocol. We clearly observe that in the default MTU case, the 802.11n protocol is able to deliver significantly higher throughput of 34.4 Mbps than the 7.5 Mbps of 802.11a/g (4.6x increase), by enabling the use of more complex and efficient modulation schemes. Moreover, we notice that only the lowest rate con-

MCS	PHY Rate (Mbps)	Modulation	FEC
9 Mbps	9	BPSK	3/4
12 Mbps	12	QPSK	1/2
18 Mbps	18	QPSK	3/4
MCS4	39	16-QAM	3/4
MCS5	52	64-QAM	2/3
MCS6	58.5	64-QAM	3/4

Table 13: Characteristics per configured MCS

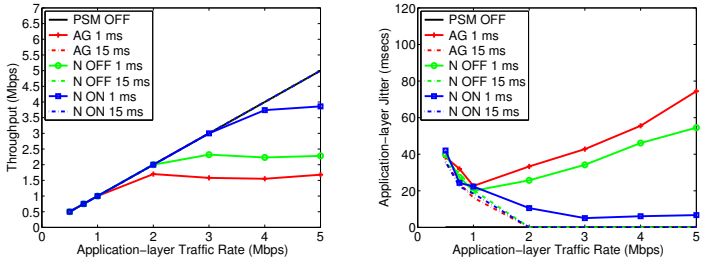
figurations of each protocol consistently provide higher throughput performance for increasing payload length, while in the rest configurations the maximum throughput is achieved under lower frame lengths. Detailed study of the throughput plots shows that proper payload length adaptation is able to provide up to 15% (8.64 Mbps - 1000 bytes payload) increase in the throughput performance of the 802.11a/g protocol and 38% (47.5 Mbps - 700 bytes payload) increase in the performance of 802.11n. The improved throughput performance is related to the increased FDR that lower payload length configurations are able to result in. In Figures 83b and 84b, we depict the FDR performance of each protocol and highlight its relation with the complexity of each modulation type.

Energy consumption measurements were also conducted, in order to evaluate the energy efficiency of each protocol across varying payload lengths. Considering the fact that the AR9380 card is characterised by a totally different power consumption profile in the two setups, along with the highly varying FDR and throughput performance, we conclude that deriving the most energy efficient payload size per case is a rather complex task. Towards deriving concrete conclusions, we plot the Effective E_B representation at the NIC level in Figures 83c and 84c for each protocol. Regarding the Effective E_B at the NIC level as obtained between the two setups, we notice that the 802.11n setup is able to reduce energy consumption down to 5.78 nJ/bit (MCS6 - 700 bytes) and offer reduction of 58% in comparison with the 13.89 nJ/bit (18 Mbps - 1000 bytes) that the 802.11a/g can offer at best. The obtained results show that payload lengths between 500 and 1200 bytes are preferable in terms of energy efficiency for the operation of the 802.11a/g protocol, while in the case of 802.11n even lower frame sizes between 300 and 1000 bytes can further reduce energy expenditure. Concluding, we remark that it is important to design automated algorithms that jointly adapt the MAC frame payload length and the PHY bit rate, towards achieving higher throughput and lower energy consumption.



(a) PSM OFF (b) PSM ON - Timeout 1 ms (c) PSM ON - Timeout 15 ms

Figure 85: Power consumption of AR9380 NIC across varying PSM configurations



(a) Throughput (b) Jitter

Figure 86: Network performance of AR9380 NIC across varying PSM configurations

9.5.3 *Experimentation with 802.11 PSM*

Through this experiment, we aim at quantifying the potential energy savings of the 802.11 PSM during periods of network activity, by experimenting in network setups that are based on both protocols under consideration. The 802.11 PSM mechanism is designed to set the wireless NICs of stations (STAs) in a low-power state during periods of inactivity and periodically activate them to fetch cached data from the access point (AP). Considering an active network, the useful period during which the STA's NIC can remain deactivated, is directly determined by the inter-packet arrival time of traffic flows that are destined to the STA. While the STA's NIC is in sleep mode, all cached frames at the AP are being delayed till the next *Beacon Interval*.

As the PSM mechanism is only able to affect the consumption of STAs, in this experiment we consider downlink transmissions and measure the impact of PSM on the energy efficiency of STAs (receivers), while also evaluating network performance metrics. We configure the high-SNR experimental setup that was described in Section 3 and equip both nodes with the AR9380 chipset that supports the 802.11 PSM mechanism. We also assign the default *Beacon Interval* of 100 ms at the AP node. Towards stressing the operation of the 802.11 PSM mechanism, we exploit the ability of the *ath9k* driver to tune the Timeout Period (*TP*) parameter, which configures the interval before the NIC goes back to sleep mode, in order to control the tradeoff between the induced delay and energy savings. In this experiment we vary the *TP* between the minimum value of

1 ms and 15 ms and investigate the impact of PSM on application-layer performance and NIC energy efficiency.

We start by measuring performance in the case that the PSM is deactivated and proceed with the next two phases, where the 1 ms and 15 ms TP intervals are configured. In each phase, we vary the application-layer traffic rate at the AP side, by configuring values between 0.5 Mbps and 5 Mbps and measure the network performance in terms of throughput and jitter, while also monitoring the energy consumption of the STA's NIC. Figures 85a, 85b and 85c present the average power consumption of the STA's NIC in each phase, considering the application of each protocol accordingly. In the case that the 802.11n protocol is applied, we also distinguish between the A-MPDU assisted (N ON) and the non-assisted case (N OFF). We observe that the 1 ms *TP* configuration provides significant energy savings, in comparison with the PSM disabled case, that approximate at maximum 74% for the 802.11a/g setup (AG) and 79% for the 802.11n setup (N ON) . Considering the 15 ms *TP* configuration, a minimal reduction of energy savings is observed, across traffic load values below 2 Mbps. In the case that the traffic load exceeds the 2 Mbps value, the NIC rarely falls in sleep mode across all the considered cases, thus resulting in lower energy savings (not exceeding 15%).

Next, we characterise the impact of PSM on application layer performance, considering throughput and jitter, as plotted in Figures 86a and 86b accordingly. In general, we observe that the 15 ms *TP* configuration poses no impact on throughput and only minimal impact on jitter performance, thus not sacrificing network performance for saving energy. On the other hand, the 1 ms *TP* configuration that stresses the operation of the 802.11 PSM mechanism, provides more interesting results that clearly highlight the impact of the 802.11n's high PHY bit rates and A-MPDU aggregation on network performance. As depicted in Fig. 86a, the increased PHY bit rates of the 802.11n improve throughput in comparison with the 802.11a/g setup, when considering traffic loads above 2 Mbps. Moreover, we observe that the application of A-MPDU aggregation further increases throughput performance, by combining several cached at the AP frames into a single A-MPDU frame, thus efficiently reducing frame losses due to buffer overflows at the transmitter. The impact of the 802.11n's increased PHY bit rates and A-MPDU aggregation in jitter performance are highlighted in Fig. 86b, where we observe that the A-MPDU assisted protocol constantly enables on time frame delivery and results in remarkably reduced jitter.

Concluding, we remark that this last experiment has clearly demonstrated that the 802.11n protocol is able to provide both increased network performance and significant energy savings through the application of the PSM mechanism, during periods of network activity. Moreover, our results have shown that scheduling of sleep intervals in

an adaptive to the prevailing traffic conditions and protocol parameters way, is able to bridge the gap between high network latency and low energy savings, as shown in [128].

9.6 CONCLUSIONS

In this work, we presented detailed experimentally obtained results that evaluate the energy efficiency of the base 802.11 standard in comparison with the latest 802.11n version, under a wide range of settings. In-depth analysis of the collected results has shown that the advanced features of the latest standard enable significant reduction of energy expenditure, across all the various considered scenarios. We envision that our findings will provide valuable insights to researchers working on the design of energy efficient wireless protocols.

10

CONCLUSIONS AND FUTURE WORK

10.1 SUMMARY OF CONTRIBUTIONS

In this thesis, we studied resource allocation algorithms in wireless networks, considering the resources of network users, the occupied spectrum and the amount of energy that is being consumed. Across all the development phases of our work, a common research approach has been followed. Particularly, all the proposed mechanisms and algorithms have been designed and implemented in real wireless devices, in order to enable performance evaluation under realistic environments and direct comparison contrary to existing standards.

10.1.1 *User Association*

We have developed a user association algorithm that jointly considers the factors affecting end-user performance in 802.11 networks. The key novelty of our approach is that it results in comprehensive metrics for user association that capture both contention from “one-hop" and interference from “two-hop" neighbors, not as individual parameters but in a joint manner. The proposed metrics consider the impact of each neighboring node proportionally to its configured PHY-layer transmission rate and its anticipated traffic activity. We verify the applicability of the proposed mechanisms by integrating them into the 802.11 protocol and evaluate their performance on commercial equipment and under realistic scenarios in a real-life testbed environment. As a first contribution, we encapsulate in our user association metrics the effect of contention. In contradiction to relevant approaches, we state that AP load should be considered over all neighboring nodes, due to the shared nature of the medium. Another key contribution of the proposed scheme is its ability to adapt to varying traffic conditions. As a third contribution we integrate the effect caused of active hidden terminals.

Our mechanism, integrating all the above features, results in algorithms proposed for both the association and handoff procedures. Through experiments, we present how users that target maximisation of their own throughput, significantly impact ag-

gregate network throughput. Inspired by these observations, we propose two discrete association strategies that target maximisation of individual user throughput and aggregate throughput performance accordingly. We show how the proposed measurement propagation procedure can be readily incorporated in the IEEE 802.11k amendment and moreover move one step further than simulation and implement the proposed algorithms using open source drivers. Through extensive testbed experiments, we show how both individual user and aggregate network throughput evolve, as the number of network nodes and injected traffic increase and thus manage to highlight the tradeoff between aggregate throughput maximisation and fair load balancing. As a first extension, we plan on studying the problem of user association and frequency selection jointly. Furthermore, a future objective is to test new mechanisms that jointly perform transmit power and association control, in order to see how the difference in transmission range can affect the “one-hop" and “two-hop" neighborhoods, which in turn affect the corresponding associations.

We also studied user association mechanisms the consider performance under combined topologies, consisting of both wireless and wired parts. In order to run realistic association experiments in the context of WMNs, we exploited the federation of the wired PLE and the wireless NITOS testbeds, which environment reproduced several characteristics of experimentation under real world scale and settings. Moreover, several testbed experiments were also reproduced in a network simulator tool, demonstrating the inability of simulated environments to accurately approximate the complex nature of the wireless medium and the high variability that experimentation over Internet generates. Through the extensive experiments and the corresponding collected results, we validated the importance of integrating experimental facilities for the design and development of the Future Internet. Moreover, we highlighted the benefits that can be offered by association mechanisms that take into account metrics able to characterize end user performance in combined topologies. As part of our future work, we plan on investigating performance of more complex WMN topologies that also feature a wireless multi-hop backhaul in the aforementioned federated environment. Furthermore, we aim at enhancing the existing tools for experimentation in federated testbeds and developing new ones, towards upgrading the perceived user experience.

10.1.2 *Spectrum Management*

We have proposed a novel dynamic frequency selection algorithm that incorporates several, hitherto unexploited features affecting total network interference and devised a distributed protocol suite that dynamically selects the operating channel. The key nov-

elty of our approach is the development of a client-assisted mechanism that enables the STAs of each BSS to participate in interference estimation. Our client feedback mechanism was extended, in order for each AP to further utilize interference measurements reported by STAs that belong to other BSSs. Another key contribution of the proposed scheme is its ability to adapt to varying traffic conditions. Thus, we were able to estimate the level of contention on each available channel along with all the fundamental issues that affect the end-user performance. Finally, another important contribution was the implementation of the proposed algorithm using open source drivers. To the best of our knowledge, our work was the first in the literature to feature a complete driver-level implementation, accompanied with such extensive experimental results.

We also extensively studied the influence of interference effects on the throughput performance of wireless networks. Moreover, by proving that wireless experiments are very susceptible to unpredictable interference, we highlighted the importance of using interference monitoring systems to validate the experimental integrity. Moreover, we designed mechanisms able to detect each type of interference, based on high-end spectrum sensing devices. We also developed a sophisticated framework that enables evaluation of cognitive devices, in terms of sensing delay and energy efficiency. We integrated the developed monitoring procedure directly in the experimentation tools of the CREW cognitive testbed and demonstrated how it aids in the online evaluation of different cognitive platforms in terms of the aforementioned metrics. In the future, we plan to integrate the developed interference detection techniques and evaluation framework into a fully automated benchmarking framework, designed for cognitive-based experimentation.

Building on top of the experience in Spectral Analysis that was gained as an outcome of our previous research efforts, we developed a Spectrum Sensing mechanism that is able to operate on commercial 802.11 hardware and scan the spectrum on an Energy detection basis. Through this approach, we enabled 802.11 compatible wireless cards to detect under-utilized spectrum fragments and properly configure both the central frequency and bandwidth of operation, in order to properly adapt to the prevailing interference and channel conditions. Our detailed experimental evaluation showcased the advantages of agile spectrum adaptation and more specifically the improved SNR and throughput under interfered conditions and the resulting decrease in energy consumption. As part of our future work, we aim at testing the developed framework across varying topologies and network configurations, in order to derive the rules that drive optimal spectrum adaptation decisions.

10.1.3 *Energy Efficiency*

We introduced a novel framework that is able to characterise the consumption of wireless infrastructure in an online way. The proposed framework is built on a distributed network of low-cost, but highly accurate devices and is able to support online monitoring of energy expenditure, along with the experiment execution. The overall advantages of the developed system are listed below:

- **Non-intrusive operation:** Our solution runs on external hardware, which does not interfere with the measured devices.
- **Online monitoring:** The proposed framework allows for online gathering of measurements in parallel with execution of long-term experiments.
- **Distributed architecture:** The proposed framework is composed of distributed communication enabled components.
- **High sampling rate:** The developed platform achieves the sampling rate of 63 KHz, which equals twice the minimum required sampling rate.
- **High sampling accuracy:** The custom developed hardware achieves accuracy in the order of milliwatts.
- **Adaptive to heterogeneous infrastructure:** The followed measurement procedure is rather generic and allows for power consumption monitoring of any device type.
- **Low-cost hardware:** The developed hardware solution introduces a total cost of less than 80.

We also enabled online power consumption monitoring of portable devices through the integration of a tiny custom-designed board, in order to provide energy efficiency evaluation of mobile devices under realistic scenarios.

Considering also the proposed power measurement methodology and using the hardware device that was introduced in the aforementioned work, we experimentally evaluated the energy efficiency of the base 802.11 standard in comparison with the latest 802.11n version, under a wide range of settings. Based on in-depth interpretation of the collected results, we highlighted the improvement in energy efficiency that the 802.11n version is able to provide. Our detailed findings can act as guidelines for researchers working on the design of energy efficient wireless protocols.

BIBLIOGRAPHY

[1] IEEE 802.11-2007 Wireless LAN Medium Access Control and Physical Layers Specifications.

[2] *"Penetration of Wi-Fi in electronic products"*, http://goo.gl/aQs6EW.

[3] *"ICT Electricity Consumption"*, http://goo.gl/FstidW.

[4] H.-O. Scheck. ICT x00026; wireless networks and their impact on global warming. In *Proceedings of the European Wireless Conference (EW), 2010*, 2010.

[5] D.C. Kilper, G. Atkinson, S.K. Korotky, S. Goyal, P. Vetter, D. Suvakovic, and O. Blume. Power Trends in Communication Networks. *Selected Topics in Quantum Electronics, IEEE Journal of*, 17(2):275–284, March 2011.

[6] N. Balasubramanian, A. Balasubramanian, and A. Venkataramani. "energy consumption in mobile phones: A measurement study and implications for network applications". In *Proceedings of IMC*, 2009.

[7] J. Tarascon. "Key challenges in future Li-battery research". *Philos Trans A Math Phys Eng Sci*, pages 3227–4, 2010.

[8] Y. Bejerano, S. Han, and E. L. Li. Fairness and load balancing in wireless LANs using association control. In *Proc. of MobiCom '04*.

[9] Y. Bejerano and Randeep S. B. MiFi: a framework for fairness and QoS assurance for current IEEE 802.11 networks with multiple access points. *Networking, IEEE/ACM Transactions on*, 2006.

[10] T. Korakis, O. Ercetin, S. Krishnamurthy, L. Tassiulas, and S. Tripathi. Link Quality based Association Mechanism in IEEE 802.11h compliant Wireless LANs. In *Proceedings of RAWNET*, 2006.

[11] 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 *Proc. of INFOCOM '07*.

[12] O. Ekici and A. Yongacoglu. A novel association algorithm for congestion relief in ieee 802.11 wlans. In *Proceedings of IWCMC*, pages 725–730, New York, NY, USA, 2006. ACM.

[13] G. Athanasiou, T. Korakis, O. Ercetin, and L. Tassiulas. Dynamic cross-layer association in 802.11-based mesh networks. In *Proceedings of INFOCOM, 2007*.

[14] K. Sundaresan and K. Papagiannaki. The need for cross-layer information in access point selection algorithms. In *Proc. of IMC '06*.

[15] M. Abusubaih and A. Wolisz. An optimal station association policy for multi-rate ieee 802.11 wireless lans. In *In Proceedings of MSWiM*, 2007.

[16] M. Abusubaih and A. Wolisz. Interference-aware decentralized access point selection policy for multi-rate ieee 802.11 wireless lans. In *Proceedings of PIMRC*, pages 1–6. IEEE, 2008.

[17] Li Li, M. Pal, and Y.R. Yang. Proportional Fairness in Multi-Rate Wireless LANs. In *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, 2008.

[18] Ka-Lok Hung, Brahim Bensaou, and Rui Li. Cross layer association control for throughput optimization in wireless LANs with inter-AP interference. In *Proceedings of the 13th ACM international conference on Modeling, analysis, and simulation of wireless and mobile systems*, MSWIM '10, pages 210–217, 2010.

[19] W. Li, S. Wang, Y. Cui, X. Cheng, R. Xin, M. A. Al-Rodhaan, and A. Al-Dhelaan. AP Association for Proportional Fairness in Multirate WLANs. *Networking, IEEE/ACM Transactions on*, PP(99):1–1, 2013.

[20] G. Bhanage, D. Vete, I. Seskar, and D. Raychaudhuri. SplitAP: Leveraging Wireless Network Virtualization for Flexible Sharing of WLANs. In *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*, pages 1–6, 2010.

[21] M. Heusse, F. Rousseau, G. Berger-Sabbatel, and A. Duda. Performance anomaly of 802.11b. In *Proc. of INFOCOM '03*.

[22] IEEE, ÒPart 11: Wireless LAN Medium Access Control (MAC) and Physical (PHY) Layer Specifications Ð Amendment 1 : Radio Resource Measurement of Wireless LANs, http://standards.ieee.org/findstds/standard/802.11k-2008.html.

[23] "Madwifi Open Source Driver", http://madwifi-project.org/.

[24] A. Kumar, E. Altman, D. Miorandi, and M. Goyal. New insights from a fixed-point analysis of single cell IEEE 802.11 WLANs. *In IEEE/ACM Trans. Netw.*, 15(3):588–601, 2007.

[25] G. Bianchi. Performance analysis of the IEEE 802.11 distributed coordination function. *In IEEE JSAC*, pages 535–547, March 2000.

[26] T. Rakotoarivelo, M. Ott, G. Jourjon, and I. Seskar. OMF: A Control and Management Framework for Networking Testbeds. *SIGOPS Oper. Syst. Rev.*, 43(4):54 –59, 2010.

[27] Rajendra K. Jain, Dah-Ming W. Chiu, and William R. Hawe. A Quantitative Measure Of Fairness And Discrimination For Resource Allocation In Shared Computer Systems. September 1984.

[28] G. Kazdaridis, S. Keranidis, A. Fiamegkos, T. Korakis, I. Koutsopoulos, and L. Tassiulas. Novel metrics and experimentation insights for dynamic frequency selection in wireless LANs. In *Proc. of Wintech '11*.

[29] I. Koutsopoulos and L. Tassiulas. Joint optimal access point selection and channel assignment in wireless networks. *IEEE/ACM Trans. Netw.*, 15(3):521–532, June 2007.

[30] Kefeng T., D. Wu, An Chan, and P. Mohapatra. Comparing simulation tools and experimental testbeds for wireless mesh networks. In *Proceedings of World of Wireless Mobile and Multimedia Networks (WoWMoM), 2010 IEEE International Symposium*.

[31] P.H. Pathak and R. Dutta. A Survey of Network Design Problems and Joint Design Approaches in Wireless Mesh Networks. *In IEEE Communications Surveys Tutorials*, 13(3):396 –428, 2011.

[32] 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 WiOpt*, 2011.

[33] Y. Bejerano, S. Han, and E. L. Li. Fairness and load balancing in wireless LANs using association control. In *Proceedings of MOBICOM*, 2004.

[34] S. Makhlouf, Ye Chen, S. Emeott, and M. Baker. A Network-Assisted Association Scheme for 802.11-Based Mesh Networks. In *Proceedings of WCNC, 2008*.

[35] L. Luo, D. Raychaudhuri, H. Liu, M. Wu, and D. Li. End-to-End Performance Aware Association in Wireless Municipal Mesh Networks. In *Proceedings of Globecom, 2009*.

[36] H. Wang, W. Wong, W. Soh, and M. Motani. Dynamic association in IEEE 802.11 based wireless mesh networks, 2009. In *Proceedings of ISWCS*.

[37] Yan He, D. Perkins, and S. Velaga. Design and Implementation of CLASS: A Cross-Layer ASSociation Scheme for Wireless Mesh Networks. In *Proceedings of INFOCOM, 2010*.

[38] C. Elliott. GENI: Opening Up New Classes of Experiments in Global Neworking. *IEEE Internet Computing,*, 14(1):39–42, 2010.

[39] A. Gavras, A. Karila, S. Fdida, M. May, and M. Potts. Future Internet research and experimentation: the FIRE initiative. *SIGCOMM Computer Communication Review,*, 37(3):89–92, 2007.

[40] Kirk Webb, Mike Hibler, Robert Ricci, Austin Clements, and Jay Lepreau. Implementing the Emulab-PlanetLab portal: Experience and lessons learned. In *Workshop on Real, Large Distributed Systems (WORLDS), 2004*.

[41] B. Chun, D. Culler, T. Roscoe, A. Bavier, L. Peterson, M. Wawrzoniak, and M. Bowman. PlanetLab: An Overlay Testbed for Broad-Coverage Services. *ACM SIGCOMM Computer Communication Review,*, 33(3):3–12, July 2003.

[42] R. Mahindra, G. Bhanage, G. Hadjichristofi, S. Ganu, P. Kamat, I. Seskar, and D. Raychaudhuri. Integration of heterogeneous networking testbeds. In *Proceedings of TridentCom 2008*.

[43] G. Di Stasi, S. Avallone, and R. Canonico. Integration of OMF-based testbeds in a global scale networking facility. In *Proceedings of ICST QShine 2009*.

[44] A. Anadiotis, A. Apostolaras, D. Syrivelis, T. Korakis, L. Tassiulas, L. Rodriguez, I. Seskar, and M. Ott. Towards maximizing wireless testbed utilization using spectrum slicing. In *Proceedings of TridentCom 2010*.

[45] G. Di Stasi, R. Bifulco, F.P. D'Elia, S. Avallone, R. Canonico, A. Apostolaras, N. Giallelis, T. Korakis, and L. Tassiulas. Experimenting with P2P traffic optimization for wireless mesh networks in a federated OMF-PlanetLab environment. In *Proceedings of WCNC, 2011*.

[46] S. Fdida, T. Friedman, and T. Parmentelat. *OneLab: An Open Federated Facility for Experimentally Driven Future Internet Research*. Springer, 2010.

[47] Panayotis Antoniadis, Serge Fdida, Timur Friedman, and Vishal Misra. Federation of virtualized infrastructures: sharing the value of diversity. In *Proceedings of the 6th International COnference*, Co-NEXT '10.

[48] *"Dummynet"*, http://info.iet.unipi.it/ luigi/dummynet/.

[49] Mathieu Lacage, Martin Ferrari, Mads Hansen, Thierry Turletti, and Walid Dabbous. NEPI: using independent simulators, emulators, and testbeds for easy experimentation. *SIGOPS Oper. Syst. Rev.*, 43(4):60–65, January 2010.

[50] "PlanetLab Central (PLC) portable installation", https://svn.planet-lab.org/wiki/MyPLCUserGuide.

[51] Jolyon White, Guillaume Jourjon, Thierry Rakotoarivelo, and Max Ott. Measurement Architectures for Network Experiments with Disconnected Mobile Nodes. In *Proceedings of TridentCom 2010, 6th International ICST Conference on Testbeds and Research Infrastructures for the Development of Networks & Communities*, 2010.

[52] "OMF Website", http://mytestbed.net/.

[53] "XMPP Protocol RFC", http://xmpp.org/rfcs/rfc6120.html.

[54] "SFA Registry", http://svn.planet-lab.org/wiki/SFATutorialRegistryInterface.

[55] "MySlice", http://myslice.info/.

[56] *"IPtables"*, http://netfilter.org.

[57] "Nmap", http://nmap.org/.

[58] *"Iperf"*, http://dast.nlanr.net/Projects/Iperf/.

[59] *"VLC Media Player"*, http://www.videolan.org/vlc/.

[60] *"Socat Relay Platform"*, http://freecode.com/projects/socat.

[61] *"NITOS-PLE Federation Experiment"*, http://nitlab.inf.uth.gr/NIT lab/images/stories/tutorials/federation.zip.

[62] *"NS3 - Network Simulator for Internet Systems"*, http://www.nsnam.org/.

[63] H. T. Friis. A Note on a Simple Transmission Formula. *Proceedings of the IRE*, (5):254–256, 2006.

[64] V. Erceg, L. J. Greenstein, S. Y. Tjandra, S. R. Parkoff, A. Gupta, B. Kulic, A. A. Julius, and R. Bianchi. An empirically based path loss model for wireless channels in suburban environments. *IEEE JSAC*, 17(7):1205–1211, 2006.

[65] M. Nakagami. The m-distribution – A general formula of intensity distribution of rapid fading. In *Statistical Methods in Radio Wave Propagation*. 1960.

[66] M. Stoffers and G. Riley. Comparing the NS-3 Propagation Models. In *Proceedings of Modeling, Analysis Simulation of Computer and Telecommunication Systems (MASCOTS), 2012 IEEE 20th International Symposium*.

[67] *"OPNET Network Simulator"*, http://www.opnet.com/.

[68] URL = http://cisco.biz/ *"Cisco Aironet", http://cisco.biz/*, key = MAD.

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

[70] 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.

[71] 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.

[72] 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.

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

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

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

[76] 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.

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

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

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

[80] *"NITOS Wireless Testbed"*, http://nitlab.inf.uth.gr/NITlab/index.php/testbed.

[81] 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.

[82] *"Ettus Research"*, http://www.ettus.com/.

[83] *"Radiotap"*, http://www.radiotap.org/.

[84] P. Sutton et al. Iris: an architecture for cognitive radio networking testbeds. *IEEE Communications Magazine*, 48(9):114–122, 2010.

[85] W. Liu et al. Real-time wide-band spectrum sensing for cognitive radio. In *Proceedings of SCVT*, 2011.

[86] *"w-ilab.t Wireless Testbed"*, http://www.crew-project.eu/wilabt.

[87] A. Ghasemi and E.S. Sousa. Spectrum sensing in cognitive radio networks: requirements, challenges and design trade-offs. *Communications Magazine, IEEE*, 46(4):32–39, April 2008.

[88] T. Yucek and H. Arslan. A survey of spectrum sensing algorithms for cognitive radio applications. *Communications Surveys Tutorials, IEEE*, 11(1):116–130, First 2009.

[89] *"Universal Software Radio Peripheral (USRP)"*, http://www.ni.com/usrp/.

[90] *"Wireless Open-Access Research Platform (WARP)"*, http://warp.rice.edu/.

[91] *"Microsoft Research Software Radio (SORA)"*, http://goo.gl/KoYkhQ.

[92] *"IMEC Sensing Engine"*, http://www2.imec.be.

[93] *"VT-CORNET"*, http://goo.gl/UTojEZ.

[94] *"Open-Access Research Testbed for Next-Generation Wireless Networks (ORBIT)"*, http://www.orbit-lab.org/.

[95] *"Emulab - Network Emulation Testbed"*, http://www.emulab.net.

[96] *"CORTEX-lab"*, http://www.cortexlab.fr/.

[97] *"Cognitive Radio Experimentation World (CREW) project"*, http://www.crew-project.eu/.

[98] *"GNURadio"*, http://gnuradio.org/.

[99] *"Software Communications Architecture (SCA)"*, http://goo.gl/EjMlQ7.

[100] *"Implementing Radio in Software (IRIS)"*, http://irissoftwareradio.wordpress.com/.

[101] S. Keranidis, G. Kazdaridis, V. Passas, T. Korakis, I. Koutsopoulos, and L. Tassiulas. Online Energy Consumption Monitoring of Wireless Testbed Infrastructure Through the NITOS EMF Framework. In *Proceedings of ACM WiNTECH*, 2013.

[102] S. Keranidis, W. Liu, M. Mehari, P. Becue, S. Bouckaert, I. Moerman, T. Korakis, I. Koutsopoulos and L.Tassiulas. CONCRETE: A Benchmarking tool to CONtrol and Classify REpeatable Testbed Experiments. Technical report, 2012.

[103] *"USRP N210 Networked Series"*, https://www.ettus.com/product/details/UN210-KIT.

[104] *"USRP E110 Embedded Series"*, https://www.ettus.com/product/details/UE110-KIT.

[105] *"Atheros AR9380 Chipset"*, http://goo.gl/51Mg7.

[106] *"Ath9k wireless driver"*, http://wireless.kernel.org/en/users/Drivers/ath9k.

[107] Wei Liu, Daan Pareit, EliDe Poorter, and Ingrid Moerman. Advanced spectrum sensing with parallel processing based on software-defined radio. *EURASIP Journal on Wireless Communications and Networking*, 2013(1), 2013.

[108] A. Mishra, V. Shrivastava, D. Agrawal, S. Banerjee, and S. Ganguly. Distributed Channel Management in Uncoordinated Wireless Environments. In *Proceedings of MOBICOM*, 2006.

[109] 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.

[110] S. Chieochan, E. Hossain, and J. Diamond. Channel assignment schemes for infrastructure-based 802.11 WLANs: A survey. *Communications Surveys Tutorials, IEEE*, 2010.

[111] G. Kazdaridis, S. Keranidis, A. Fiamegkos, T. Korakis, I. Koutsopoulos, and L. Tassiulas. Novel Metrics and Experimentation Insights for Dynamic Frequency Selection in Wireless LANs. In *Proceedings of ACM WiNTECH*, 2011.

[112] Shravan Rayanchu, Ashish Patro, and Suman Banerjee. Airshark: Detecting non-WiFi RF Devices Using Commodity WiFi Hardware. In *Proceedings of IMC*, 2011.

[113] S. Gollakota, F. Adib, D. Katabi, and S. Seshan. Clearing the RF Smog: Making 802.11N Robust to Cross-technology Interference. In *Proceedings of SIGCOMM*, 2011.

[114] IEEE 802.11n-2009, Amendment 5: Enhancements for Higher Throughput., 2009.

[115] IEEE 802.11ac-2013, Amendment 4: Enhancements for very high throughput for operation in bands below 6 GHz., 2009.

[116] Ranveer Chandra, Ratul Mahajan, Thomas Moscibroda, Ramya Raghavendra, and Paramvir Bahl. A Case for Adapting Channel Width in Wireless Networks. *SIGCOMM Comput. Commun. Rev.*, 38(4), aug 2008.

[117] H. Rahul, N. Kushman, D.a Katabi, C. Sodini, and F. Edalat. Learning to Share: Narrowband-friendly Wideband Networks. In *Proceedings of SIGCOMM*, 2008.

[118] L. Yang, W. Hou, L. Cao, B. Y. Zhao, and H. Zheng. Supporting Demanding Wireless Applications with Frequency-agile Radios. In *Proceedings of NSDI*, 2010.

[119] Krishna C., Bozidar R., Vlad B., Michael B., Srinivas Yerramalli, Vishnu Navda, and Ramachandran Ramjee. WiFi-NC : WiFi Over Narrow Channels. In *Proceedings of NSDI*, 2012.

[120] S. Yun, D. Kim, and L. Qiu. Fine-grained Spectrum Adaptation in WiFi Networks. In *Proceedings of MOBICOM*, 2013.

[121] S. Rayanchu, V. Shrivastava, S. Banerjee, and R. Chandra. FLUID: Improving Throughputs in Enterprise Wireless Lans Through Flexible Channelization. In *Proceedings of MOBICOM*, 2011.

[122] X. Zhang and K. Shin. "EMiLi: energy-minimizing idle listening in wireless networks". In *Proceedings of Mobicom*, 2011.

[123] Kishore R., Ravi K., Honghai Z., and Marco G. "Symphony: Synchronous Two-phase Rate and Power Control in 802.11 WLANs". In *Proceedings of Mobisys*, 2008.

[124] J. Manweiler and R. R. Choudhury. "Avoiding the rush hours: WiFi energy management via traffic isolation". In *Proceedings of MobiSys*, 2011.

[125] M. Ra, J. Paek, A. B. Sharma, R. Govindan, M. H. Krieger, and M. J. Neely. "Energy-delay tradeoffs in Smartphone Applications". In *Proceedings of MobiSys*, 2010.

[126] *"Tmote sky Specifications"*, http://goo.gl/Tc9qR.

[127] *"MICAz Specifications"*, http://goo.gl/rLYHU.

[128] K. Jang, S. Hao, A. Sheth, and R. Govindan. "Snooze: energy management in 802.11n WLANs". In *Proceedings of CoNEXT*, 2011.

[129] C. Li, C. Peng, S. Lu, and X. Wang. "Energy-based rate adaptation for 802.11n". In *Proceedings of Mobicom*, 2012.

[130] D. Halperin, B. Greenstein, A. Sheth, and D. Wetherall. Demystifying 802.11n power consumption. In *Proceedings of SIGOPS HotPower*, 2010.

[131] A. Hergenroder, J. Horneber, and J. Wilke. "SANDbed: A WSAN Testbed for Network Management and Energy Monitoring". In *GIITG KuVS Sensornetze*, 2009.

[132] A. Kipp, J. Liu, T. Jiang, J. Bucholz, L. Schubert, M. Berge, and W. Christmann. "Testbed architecture for generic, energy-aware evaluations and optimisations". In *Infocomp*, 2011.

[133] K. Gomez, R. Riggio, T. Rashed, D. Miorandi, and F. Granelli. "Energino: Hardware and Software Solution for Energy Consumption Monitoring". In *Proceedings of WiOpt*, 2012.

[134] G. Kazdaridis, S. Keranidis, H. Niavis, T. Korakis, I. Koutsopoulos, and L. Tassiulas. "An Integrated Chassis Manager Card Platform Featuring Multiple Sensor Modules". In *Proceedings of Tridentcom*, 2012.

[135] *"ATmega 2560 micro-controller"*, http://goo.gl/IFHwq.

[136] *"Arduino Ethernet Shield"*, http://goo.gl/LXs10G.

[137] *"Texas Instruments INA139"*, http://goo.gl/rPQLB.

[138] *"Advanced Arduino ADC"*, http://goo.gl/AwQ95.

[139] *"Enhancing Arduino's ADC"*, http://goo.gl/BRXCX.

[140] *"Atmega ADC accuracy vs clock speed"*, http://goo.gl/qTlhx.

[141] *"NI-6210 DAQ module"*, http://goo.gl/oFSJw.

[142] *"Energy Characteristics of NITOS NICs"*, http://nitlab.inf.uth.gr/NITlab/papers/EnergyTR.pdf.

[143] M. Tauber and S.N. Bhatti. "The Effect of the 802.11 Power Save Mechanism (PSM) on Energy Efficiency and Performance during System Activity". In *Proceedings of GreenCom*, 2012.

[144] *"Internet Control Message Protocol"*, http://goo.gl/bjQCyr.

[145] J. Lee, W. Kim, S. Lee, D. Jo, J. Ryu, T. Kwon, and Y Choi. "An experimental study on the capture effect in 802.11a networks". In *ACM WinTECH*, 2007.

[146] *"Pro Micro Board"*, http://goo.gl/9QdRs8.

[147] *"ATmega 32u4 micro-controller"*, http://goo.gl/gln5Fp.

[148] *"RN-42N Bluetooth Radio"*, http://goo.gl/6MRgiH.

[149] D. Skordoulis, Qiang Ni, Hsiao-Hwa Chen, A.P. Stephens, Changwen Liu, and A. Jamalipour. IEEE 802.11n MAC frame aggregation mechanisms for next-generation high-throughput WLANs. *Wireless Communications, IEEE*, 15(1):40–47, 2008.

[150] K. Pelechrinis, T. Salonidis, H. Lundgren, and N. Vaidya. Experimental characterization of 802.11n link quality at high rates. In *Proceedings of ACM WiNTECH*, 2010.

[151] L. Kriara, M.K. Marina, and A. Farshad. Characterization of 802.11n wireless LAN performance via testbed measurements and statistical analysis. In *Proceedings of SECON*, 2013.

[152] I. Pefkianakis, Chi-Yu L., and Songwu L. What is wrong/right with IEEE 802.11n Spatial Multiplexing Power Save feature? In *Proceedings of ICNP*, 2011.

[153] G. Bhanage, R. Mahindra, I. Seskar, and D. Raychaudhuri. Implication of MAC frame aggregation on empirical wireless experimentation. In *Proceedings of GLOBECOM*, 2009.

[154] Y. Xiao and J. Rosdahl. *Throughput and delay limits of IEEE 802.11, IEEE Communications Letters, 2002*.

[155] *"Vendor Specific Improvements"*, http://goo.gl/ti8cxT.

[156] *"Wireless Incompatibilities"*, http://goo.gl/5c1cHq.

[157] IEEE 802.11n-2009, Amendment 5: Enhancements for Higher Throughput., 2009.

[158] B. Ginzburg and Kesselman A. Performance analysis of A-MSDU andA-MPDU aggregation in IEEE 802.11n. In *Proceedings of IEEE SARNOFF*, 2007.

[159] *"Maximal-ratio combining"*, http://goo.gl/ABj8PY.