

**Πανεπιστήμιο Θεσσαλίας**

**Μελέτη και υλοποίηση αλγορίθμων αναγνώρισης φάσματος με χρήση τεχνικών  
αναγνώρισης ενέργειας σε εμπορικές συσκευές**

του

**Κωνσταντίνου Χ. Χούνου**

Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

Μεταπτυχιακή Διατριβή

Επιστήμη και Τεχνολογία Υπολογιστών, Τηλεπικοινωνιών & Δικτύων



**Οκτώβριος 2014**

Επιτροπή Διατριβής:  
Λέκτορας Κοράκης Αθανάσιος  
Καθηγητής Τασιούλας Λέανδρος  
Λέκτορας Αργυρίου Αντώνιος

**University of Thessaly**

**Study and implementation of spectral sensing algorithms using energy detection techniques on commercial devices**

by

**Konstantinos C. Chounos**

Department of Electrical and Computer Engineering

Master Thesis

Science of Computer, Telecommunications & Networking Engineering



**October 2014**

Dissertation Committee:  
Lecturer Korakis Athanasios  
Prof. Tassiulas Leandros  
Lecturer Antonios Argyriou

## Περίληψη

Σε αυτή την πτυχιακή αναπτύχθηκε μια μέθοδος εκτίμησης του ποσοστού κατάληψης του ασύρματου μέσου η οποία μπορεί να υλοποιηθεί άμεσα σε εμπορικές συσκευές. Πιο συγκεκριμένα η εργασία ξεκινάει μελετώντας τους διαθέσιμους στην βιβλιογραφία αλγορίθμους ανίχνευσης φάσματος. Από αυτούς επιλέχθηκε η πιο γενικευμένη μέθοδος της ανίχνευσης ενέργειας, πάνω στην οποία βασίζεται και ο τρόπος πρόσβασης στο μέσο όπως τον υλοποιεί το πρωτόκολλο IEEE 802.11. Έπειτα μελετήθηκε η πληροφορία φάσματος που μπορεί να εξάγει μια εμπορική 802.11 ασύρματη κάρτα και στο επίπεδο του ανοίχτου λογισμικού οδηγού αυτής, υλοποιήθηκε ο αλγόριθμος μας. Δεδομένου ότι ο αλγορίθμος βασίζεται στην γενική μέθοδο της ανίχνευσης ενέργειας του φάσματος, έχουμε την δυνατότητα να εκτιμούμε τόσο μεταδόσεις που προκαλούνται από IEEE 802.11 συσκευές όσο και οποιασδήποτε μορφής μετάδοσης πραγματοποιείται στην ίδια περιοχή συχνοτήτων. Μέσα από πειραματικές μετρήσεις παρουσιάζεται πως οι συμβατικές ασύρματες κάρτες μπορούν να παρέχουν δυνατότητες ελέγχου φάσματος και αποφυγής παρεμβολών, αντίστοιχες με υψηλών προδιαγραφών συσκευές.

## **Abstract**

In this master thesis a method of estimating the medium access occupancy was developed which can be directly implemented on commercial devices. More specifically it begins by studying the available on the literature spectrum sensing algorithms. From these spectrum algorithms the most general method of energy detection was selected, on which is based medium access in the 802.11 protocol. Then the spectral information which can be extracted from a commercial 802.11 device were studied We implemented our algorithm at the level of the open source driver. Since the algorithm is based on the general method of energy detection, we are able to estimate both transmissions caused by IEEE 802.11 devices and any form of energy transmissions, within the same frequency range. Through experimental results is shown how the commercial wireless cards can provide control capabilities of the spectrum and interference avoidance, corresponding with high devices.

## Ευχαριστίες

Με την περάτωση αυτής της μεταπτυχιακής εργασίας θα ήθελα να ευχαριστήσω θερμά τον Λέκτορα του Τμήματος Ηλεκτρολόγων Μηχανικών & Μηχανικών Η/Υ κ. Κοράκη Αθανάσιο όπως και τον Καθηγητή κ. Λέανδρο Τασιούλα για την υποστήριξη που μου προσέφερε πριν αλλά και κατά την διάρκεια της διπλωματικής αυτής. Θεωρώ εξαιρετικό προνόμιο να ανήκω στην ερευνητική ομάδα του NITLAB.

Επίσης θα ήθελα να ευχαριστήσω ξεχωριστά όλη την ερευνητική ομάδα του NITLAB και ιδιαίτερα τον διδάκτορα του Τμήματος κ. Στράτο Κερανίδη, όπως και τους Ηλία Συρίγο και Βιργίλιο Πασσά για την πολύτιμη καθημερινή βοήθεια τους.

Τέλος ευχαριστώ πολύ την οικογενειά μου για την συμπαράσταση όλων αυτών των χρόνων.

## Table Of Contents

Chapter	Page
Περίληψη .....	iii
Abstract .....	iv
Ευχαριστίες .....	v
Table Of Contents .....	vi
List Of Tables .....	viii
List Of Figures .....	ix
Chapter I: Introduction.....	1
Problem Statement .....	1
Importance of the research.....	2
Structure of the Thesis .....	2
Chapter II: Theoretical Foundation.....	3
Spectrum Sensing Techniques .....	5
Energy Detection .....	5
Cyclostationary Feature Detection.....	6
Matched Filter Detection .....	7
Spectrum Sensing Techniques Comparison.....	7
Typical Cognitive Devices.....	9
Software Defined Radio.....	9
Commercial Devices as Cognitive.....	10
Chapter III: Spectrum Sensing on Commercial Devices .....	11
Atheros Wireless Adapters .....	11
Intel Wireless Adapters.....	13
Cisco Spectral Analysis .....	14
Chapter IV: Algorithm and Implementation.....	16
Default Spectral Scan Implementation in Ath9k Driver.....	16
Proposed Algorithm .....	17
Retrieving Spectral Measurements .....	17
Calculation of RSSI Statistics.....	18
Metric Definition .....	20
Adaptation to varying bandwidth capabilities .....	21
Normalize the Channel Statistics Over Time.....	22
Calculation of Optimal Frequencies .....	22
Implementation .....	24
Algorithm.....	24
Driver Modifications.....	24
Graphs.....	25
Chapter V: Experiments.....	26
IEEE 802.11 Devices .....	26
5GHz IEEE 802.11 Interferer .....	26
Non IEEE 802.11 Devices .....	29
Microwave Oven.....	29

Quadcopter Radio Controller .....	30
Bluetooth.....	30
Wireless Camera .....	31
Chapter VI: Conclusions.....	32
References.....	33

## List Of Tables

Table	Page
Table 1: Scenario Results.....	26
Table 2: Duty Cycle Results .....	27



## List Of Figures

Figure	Page
Figure 1: Spectrum utilization measurements.....	1
Figure 2: Cognitive Cycle.....	3
Figure 3: Cognitive Radio concept [4].....	4
Figure 4: Various aspects of spectrum sensing for cognitive radio.....	5
Figure 5: Implementation of an energy detector.....	6
Figure 6: Implementation of a cyclostationary detector.....	7
Figure 7: Spectrum sensing algorithms comparison.....	8
Figure 8: SDR typical architecture [7].....	10
Figure 9: Spectrogram.....	13
Figure 10: Atheros PSD graph.....	13
Figure 11: CSI tool.....	14
Figure 12: Cisco Spectral analysis visualization tools.....	15
Figure 13: Spectral scan procedure overview.....	16
Figure 14: Proposed algorithm.....	17
Figure 15: Exported data.....	18
Figure 16: Number Of Generated Samples.....	19
Figure 17: Raw data averaging.....	20
Figure 18: Duty cycle percentage calculation.....	21
Figure 19: Results matrix.....	22
Figure 20: Optimal channels selection.....	23
Figure 21: Interferer Duty Cycle.....	28
Figure 22: Interferer RSSI.....	28
Figure 22: IEEE 802.11 transmit mask[19].....	29
Figure 24: Microwave Oven RSSI and Duty Cycle.....	29
Figure 25: Quadcopter R/C RSSI and Duty Cycle.....	30
Figure 26: Bluetooth RSSI and Duty Cycle.....	31
Figure 27: Wireless Camera RSSI and Duty Cycle.....	31

# Chapter I: Introduction

## Problem Statement

The last decade the usage of mobile devices which are always connected to various wireless services increased sharply. The conventional approach to spectrum management is very inflexible. Gradually this resulted to the reduction of the free spectrum for transmissions. Nowadays in the frequencies between 30 MHz and 3GHz, the opportunities for transmissions are nearly equal to zero. However very small percentage of these frequencies is used enough over time. Figure 1 depicts the aforementioned phenomenon[1].

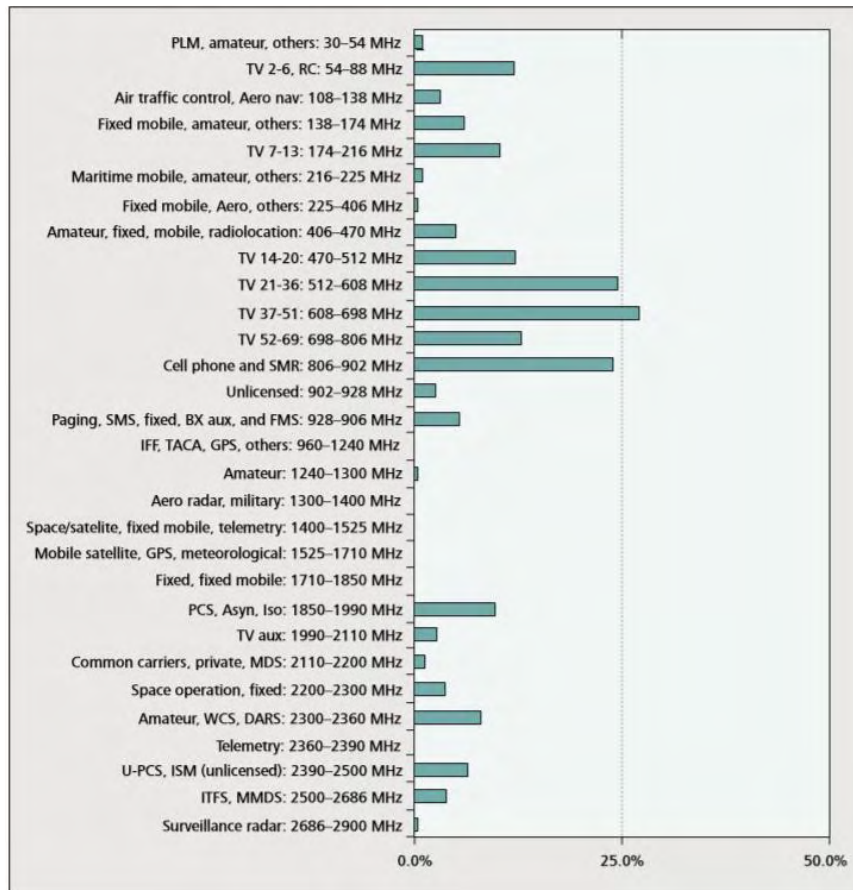


Figure 1: Spectrum utilization measurements.

Unlicensed bands (ISM) face the problem of spectrum scarcity strongly too. In particular the 2.4GHz appears to be the most used spectrum band. The coexistence of multiple

wireless technologies in a small piece of 100 MHz generates huge problems in the transmissions. Indicatively

- IEEE 802.11
- ZigBee
- Bluetooth
- Cordless Phones
- Security Cameras

are some of the technologies which are using the 2.4GHz band. Beside the above technologies which transmit under the rules of a given standard, there are other that transmit without planning. A typical example of the second category is the microwave ovens.

### ***Importance of the research***

The strong presence of the spectrum scarcity problem led academics and industry searching for suitable solutions. Cognitive Radio (CR) networks are studied for a long time and it seems that solve the problem. Through these years many architectures and techniques, proposed for the creation of the most effective cognitive radio network. These networks can be implemented under varying equipment.

### ***Structure of the Thesis***

This master thesis structured as follows. In the second chapter the basic techniques and mechanisms of the Cognitive Radios will be analyzed. In the third chapter a review of the available commercial Cognitive-enabled devices will be placed. The fourth chapter contains the proposed algorithm and its implementation. Then the experimental results from several scenarios are listed. The last chapter states the conclusions of this master thesis.

## Chapter II: Theoretical Foundation

The idea of Cognitive Radio was first proposed by Joseph Mitola. Many definitions for Cognitive Radio have been formulated from time to time. Summarizing these:

***“Cognitive radio (CR) is an intelligent radio system which has the ability to understand the current spectrum conditions and reconfigure its transmission-reception parameters in real time. “***

Such parameters indicatively can be the frequency band, the transmission power and the modulation of the transmitted signal. A cognitive radio system should be:

- Adaptive (Capability to alter operational characteristics)
- Aware (interpreted understanding of input data)
- Autonomous (not requiring user intervention)

When cognitive radio is proposed, an intelligent communication technology is expected, including observe, orient, plan, learn, decide and act. (Mitola & Maguire, 1999; Haykin 2005). The basic idea of the initial cognitive cycle is depicted in Figure 2 [2].

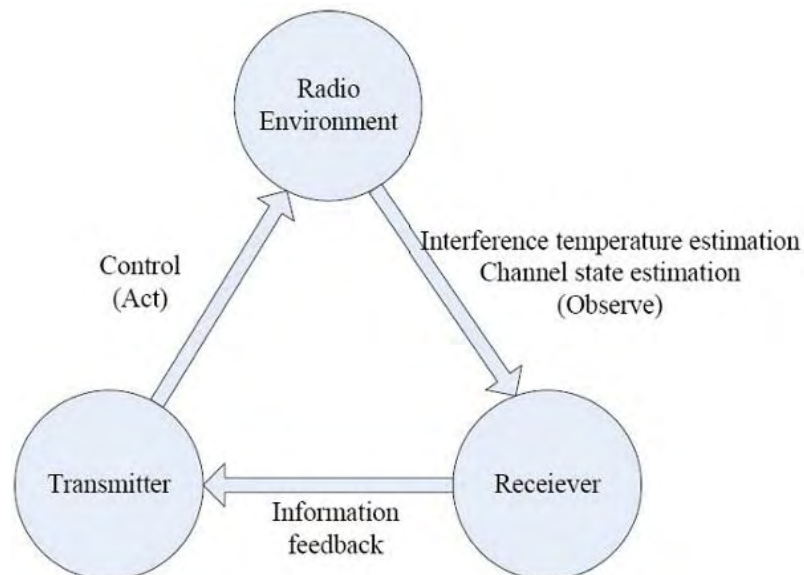


Figure 2: Cognitive Cycle.

Cognitive Radio networks can be applied in the three following types of bands:

- Licensed Bands
- Unlicensed Bands (ISM)
- Opportunistic Unlicensed Bands (TV white Spaces)

Abstractly we can distinguish the three main entities by which a cognitive radio system is composed:

- Licensed / Primary Users
- Unlicensed / Secondary Users
- Spectrum hole / white space

Overall in a cognitive radio system, the secondary user must be able, through some procedures, to diagnose the current conditions of the spectrum and reconfigure its parameters dynamically. The more efficient the previous procedure is the less interference between primary and secondary user will exist. Proper function of cognitive radio system may result better spectral efficiency and power saving of the system nodes[6].

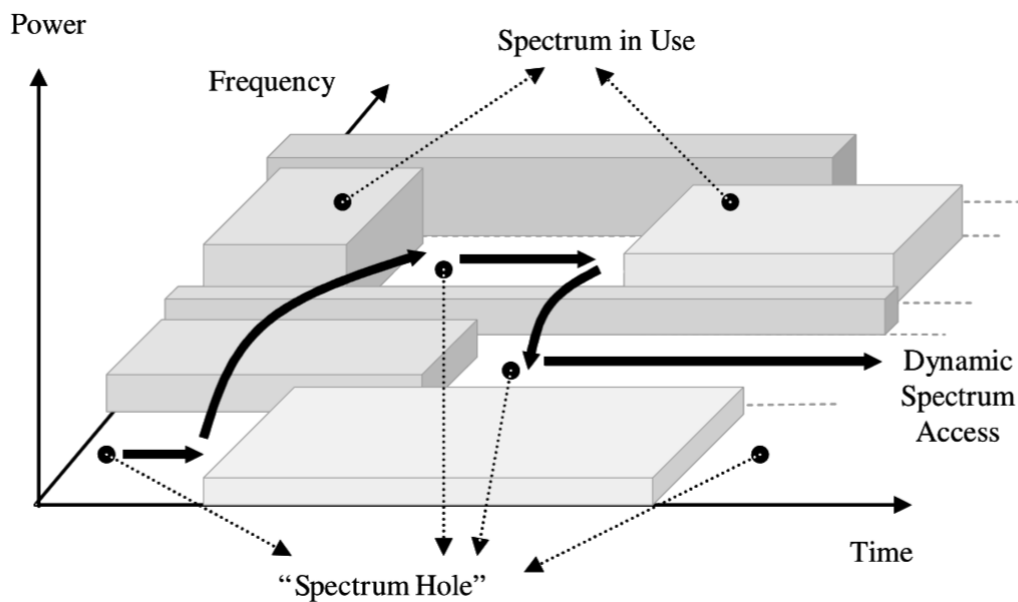


Figure 3: Cognitive Radio concept [4].

Figure 3 depicts the desired functionality of a Cognitive Radio system. For achieving this the appropriate devices have to be used. Below is given a brief description of typical Cognitive devices.

## Spectrum Sensing Techniques

Spectrum Sensing can be characterized as the main process which is responsible for keeping aware the secondary user about the current spectral conditions. There are many approaches for performing Spectrum Sensing in Cognitive Radios. Figure 4 [3] illustrates the methods of Spectrum Sensing and some other things are associated with this.

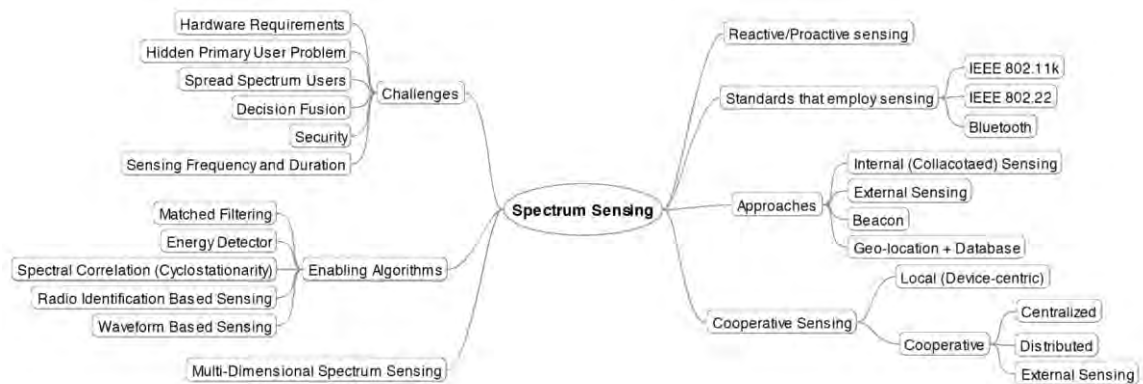


Figure 4: Various aspects of spectrum sensing for cognitive radio.

## Energy Detection

Energy detector based approach, also known as radiometry or periodogram, is the most common way of spectrum sensing because of its low computational and implementation complexities. In addition, it is more generic as receivers do not need any knowledge on the primary user's signal. The signal is detected by comparing the output of the energy detector with a threshold which depends on the noise floor [3].

There are several drawbacks of energy detectors. First, the threshold which used for primary user detection is highly susceptible to unknown or changing noise levels. Even if the threshold would be set adaptively, presence of any in-band interference would confuse the energy detector. Furthermore, in frequency selective fading it is not clear how to set the threshold with respect to channel notches. Second, energy detector does not differentiate between modulated signals, noise and interference. Since, it cannot recognize the interference, it cannot benefit from adaptive signal processing for canceling the interferer. Furthermore, spectrum policy for using the band is constrained only to primary users, so a cognitive user should treat noise and other secondary users differently.

An energy detector can be implemented similar to a spectrum analyzer by averaging frequency bins of a Fast Fourier Transform (FFT), as outlined in Figure 5. Processing gain is proportional to FFT size  $N$  and observation/averaging time  $T$ . Increasing  $N$  improves frequency resolution which helps narrowband signal detection. Also, longer averaging time reduces the noise power thus improves SNR. However, due to non-coherent processing  $O(1/\text{SNR}^2)$  samples are required to meet a probability of detection constraint [5].

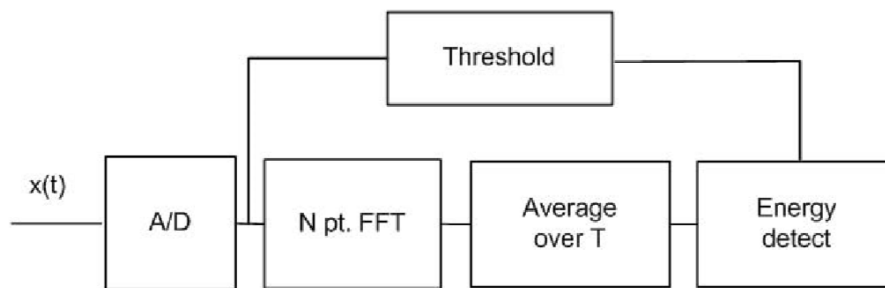


Figure 5: Implementation of an energy detector.

## Cyclostationary Feature Detection

Modulated signals are in general coupled with sine wave carriers, pulse trains, repeating spreading, hopping sequences, or cyclic prefixes which result in built-in periodicity. Even though the data is a stationary random process, these modulated signals are characterized as cyclostationary, since their statistics, mean and autocorrelation, exhibit periodicity. This periodicity is typically introduced intentionally in the signal format so that a receiver can exploit it for: parameter estimation such as carrier phase, pulse timing, or direction of arrival. This can then be used for detection of a random signal with a particular modulation type in a background of noise and other modulated signals. Common analysis of stationary random signals is based on autocorrelation function and power spectral density. On the other hand, cyclostationary signals exhibit correlation between widely separated spectral components due to spectral redundancy caused by periodicity. By analogy with the definition of conventional autocorrelation, one can define spectral correlation function (SCF).

Unlike PSD which is real-valued one dimensional transform, the SCF is two dimensional transform, in general complex-valued and the parameter  $\alpha$  is called cycle frequency. Power spectral density is a special case of a spectral correlation function for  $\alpha=0$ . The distinctive character of spectral redundancy makes signal selectivity possible. Signal

analysis in cyclic spectrum domain preserves phase and frequency information related to timing parameters in modulated signals. As a result, overlapping features in the power spectrum density are non-overlapping feature in the cyclic spectrum. Different types of modulated signals (such as BPSK, QPSK, SQPSK) that have identical power spectral density functions can have highly distinct spectral correlation functions. Furthermore, stationary noise and interference exhibit no spectral correlation [5].

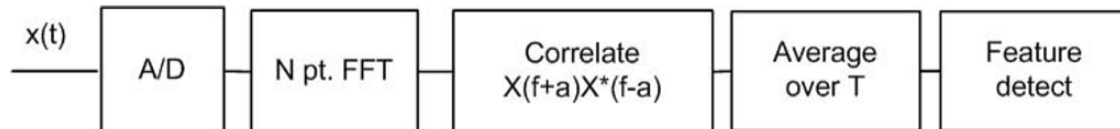


Figure 6: Implementation of a cyclostationary detector.

### Matched Filter Detection

The optimal way for any signal detection is a matched filter, since it maximizes received signal-to-noise ratio. However, a matched filter effectively requires demodulation of a primary user signal. This means that cognitive radio has full priori knowledge of primary user signal at both PHY and MAC layers, e.g. modulation type and order, pulse shaping, packet format. Such information might be pre-stored in CR memory, but the cumbersome part is that for demodulation it has to achieve coherency with primary user signal by performing timing and carrier synchronization, even channel equalization. This is still possible since most primary users have pilots, preambles, synchronization words or spreading codes that can be used for coherent detection. For examples: TV signal has narrowband pilot for audio and video carriers; CDMA systems have dedicated spreading codes for pilot and synchronization channels; OFDM packets have preambles for packet acquisition. The main advantage of matched filter is that due to coherency it requires less time to achieve high processing gain since only  $O(1/\text{SNR})$  samples are needed to meet a given probability of detection constraint. However, a significant drawback of a matched filter is that a cognitive radio would need a dedicated receiver for every primary user class[5].

### Spectrum Sensing Techniques Comparison

Beside the three aforementioned spectrum sensing techniques there are some more available in the literature. However these three seem to be the most widely used. In figure 7 [3] there is a graphical comparison of the available spectrum sensing techniques.



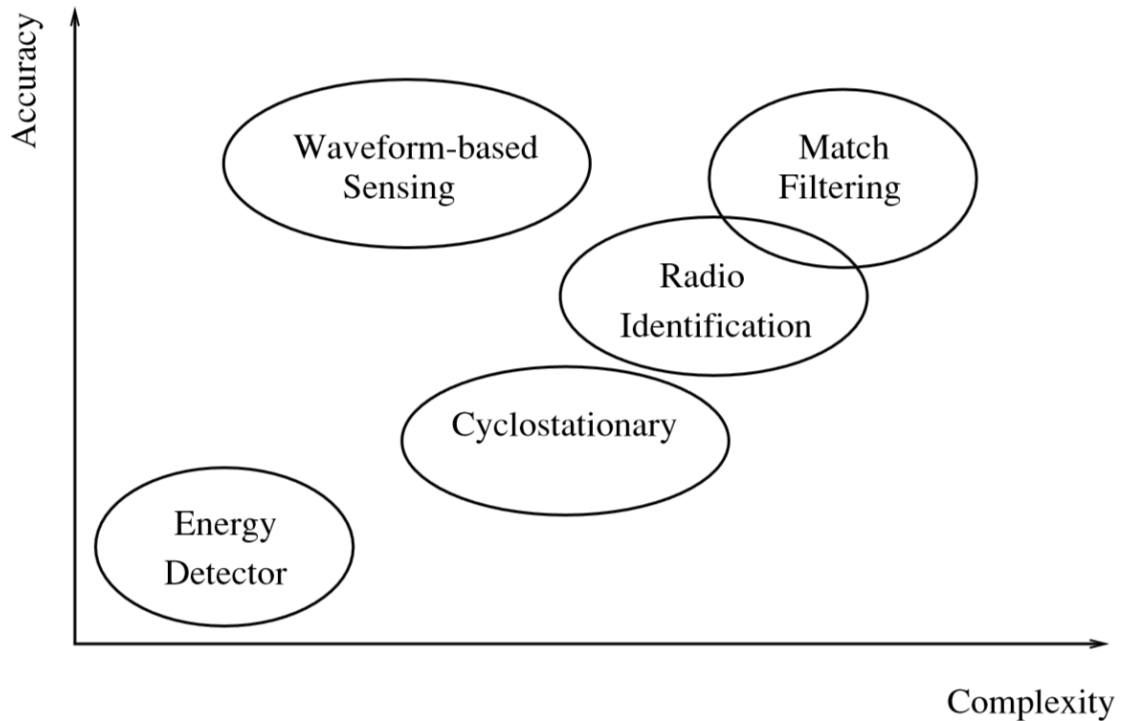


Figure 7: Spectrum sensing algorithms comparison

In summary the positives and negatives of the three aforementioned spectrum sensing algorithms are mentioned.

### Energy Detection

- **Pros**
  - No primary signal information is needed
  - Very simple implementation
  - Low computational complexity
  - Short sensing time
- **Cons**
  - No differentiation between primary and secondary users
  - Poor performance when SNR is low

### Cyclostationary Feature Detection

- **Pros**
  - Performs better than Energy Detection in low SNR
  - Capable of detecting different types of signals
- **Cons**

- Higher computational complexity than the Energy Detection
- Long sensing time

### **Matched Filter Detection**

- **Pros**
  - Short sensing time
  - Higher probability of detection than the other spectrum sensing algorithms
- **Cons**
  - Full prior knowledge of the signal need to be sensed
  - Capable of detecting only one type of signal

Considering the above parameters and carefully studding over each communications system, we can select the appropriate method of spectrum sensing.

## ***Typical Cognitive Devices***

### **Software Defined Radio**

As mentioned above the Cognitive Radio networks can be implemented using several hardware, technologies and tools, however widely used are Software Defined Radios (SDR), in which typical components (Analog Radio Frequency, A/D and D/A converters, Digital down Converters, Data Communication Interfaces.) have been implemented in hardware. In this case the whole radio system is implemented through software using the generic hardware components. Typical reasons for building Cognitive Radio systems with SDR:

- Through the generic hardware given by these radios, multiple and new standards may be implemented
- Zero upgrade cost, all changes are software based
- Problems may occur more often in hardware parts in contrast to software

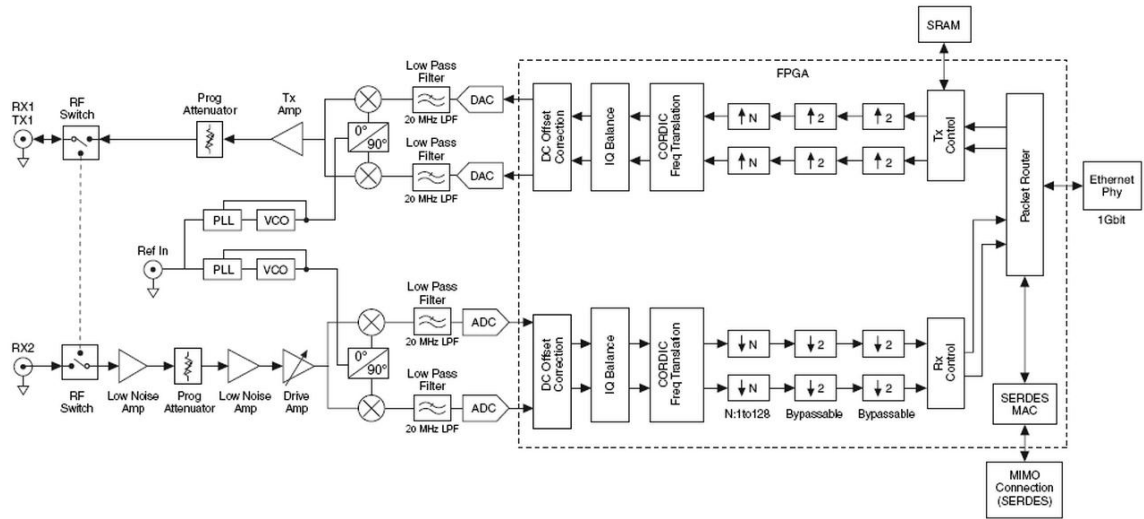


Figure 8: SDR typical architecture [7]

Typical example of SDR device is USRP N210. The USRP N210 provides high-bandwidth, high-dynamic range processing capability. The product architecture includes a Xilinx® Spartan® 3A-DSP 3400 FPGA, 100 MS/s dual ADC, 400 MS/s dual DAC and Gigabit Ethernet connectivity to stream data to and from host processors. The USRP N210 can stream up to 50 MS/s to and from host applications. Users can implement custom functions in the FPGA fabric, or in the on-board 32-bit RISC softcore. The FPGA also offers the potential to process up to 100 MS/s in both the transmit and receive directions [8]. Great flexibility is given by those devices through the interchangeable daughterboards which support frequencies from 1 MHz to 5GHz. This offers the opportunity for experimenting in multiple standards.

### Commercial Devices as Cognitive

Modern trends tend to make Cognitive Radio Networks more applicable in daily used devices. There are several wireless commercial adapters - routers which embed Cognitive Network functions. Some wireless adapters of Atheros and Intel may export data which are useful for being aware of the spectrum conditions. Chapter III include a description of these devices. This master thesis focuses on implementing Spectrum Sensing and interference avoidance algorithm based on Atheros wireless adapters.

## Chapter III: Spectrum Sensing on Commercial Devices

### *Atheros Wireless Adapters*

The Atheros AR9280 and later NICs have the ability to report FFT data from the baseband. Combing this feature with the use of an open source driver, we can retrieve useful information of the spectrum conditions. These information include:

- absolute magnitude ( $|i|+|q|$ , `abs()` for I/Q phase of the wireless signal) for each FFT bin (56 for subcarriers in HT20 mode and 128 in HT40 mode)
- an index indicating the strongest FFT bin
- the maximum signal magnitude for each sample

The function of spectral scan can be triggered from an external bash script which typically looks like:

```
echo chanscan > /sys/kernel/debug/ieee80211/phy0/ath9k/spectral_scan_ctl
iw dev wlan0 scan
cat /sys/kernel/debug/ieee80211/phy0/ath9k/spectral_scan0 > samples
echo disable > /sys/kernel/debug/ieee80211/phy0/ath9k/spectral_scan_ctl
```

In the older versions of (compat ath9k) a driver patch had to be installed. In the current versions of (backports ath9k) this patch is already into the source code. The driver creates several proc files, which are the communication point between user and driver. Some of the configurable parameters are:

- `spectral_count`: number of scan results requested.
- `spectral_short_repeat`: controls the `short_repeat` parameter (controls whether the chip is in spectral scan mode for 4 usec (enabled) or 204 usec (disabled))
- `spectral_fft_period`: controls the `fft_period` parameter (when active and triggered, PHY passes FFT frames to MAC every  $(fft\_period+1)*4\mu S$ )
- `spectral_period`: controls the `period` parameter (when active, time period between successive spectral scan entry points ( $period*256*Tclk$ ).  $Tclk = 44$  MHz for HT20 operation, 88 MHz for HT40 operation)
- `spectral_scan_ctl`: Contains the current mode. There are the following modes available:
  - `disable`: spectral scan is disabled
  - `background`: spectral scans samples are returned endlessly from the currently configured channel. It is running while the hardware is not busy

- with sending/receiving. Must be turned on by writing "trigger" into spectral\_scan\_ctl.
- manual: as many spectral scan samples as configured in spectral\_count are returned from the current channel after writing "trigger" into spectral\_scan\_ctl.
  - chanscan: as many spectral scan samples as configured in spectral\_count are returned for each channel when performing a scan.
  - spectral\_scan0: this is the file which returns the spectral scan samples. The samples are returned as TLV binary data

The frame format is:

	0	1	2	3	4	5	6	7	8
0		[7:0]: bin -28	magnitude	( i  +  q )	>>	max_exp			
1		[7:0]: bin -27	magnitude	( i  +  q )	>>	max_exp			
2-54									
55		[7:0]: bin 27	magnitude	( i  +  q )	>>	max_exp			
56		[7:0]: all_bins	{max_magnite[1:0],	bitmap_weight[5:0]}					
57		[7:0]: all_bins	{max_magnite[9:2]}						
58		[7:0]: all_bins	{max_index[5:0],	max_magnite[11:10]}					
59			[3:0]	max_exp					

Assuming the noise floor is equal to -96dbm(\*) and the magnitude of each sample in a 20 MHz bin equals the RSSI, the received signal strength of each FFT bin on HT20 channel can be computed as follow:

$$\text{power}(i) = \text{nf} + \text{RSSI} + 10 \cdot \log(\text{b}(i)^2) - \text{bin\_sum}$$

where:

- RSSI is computed on control chain 0
- b(i) is the magnitude in each bin, unscaled by max\_exp
- bin\_sum = 10\*log(sum[i=1..56](b(i)^2)) [9].

Researchers use Power Spectral Density (PSD) and Spectrogram plots in order to understand better the collected FFT data.

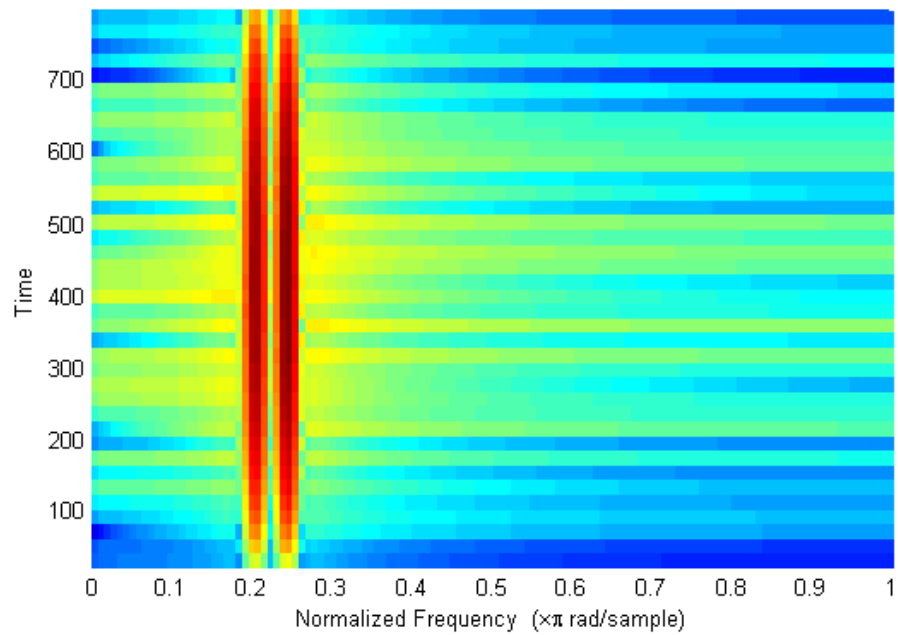


Figure 9: Spectrogram

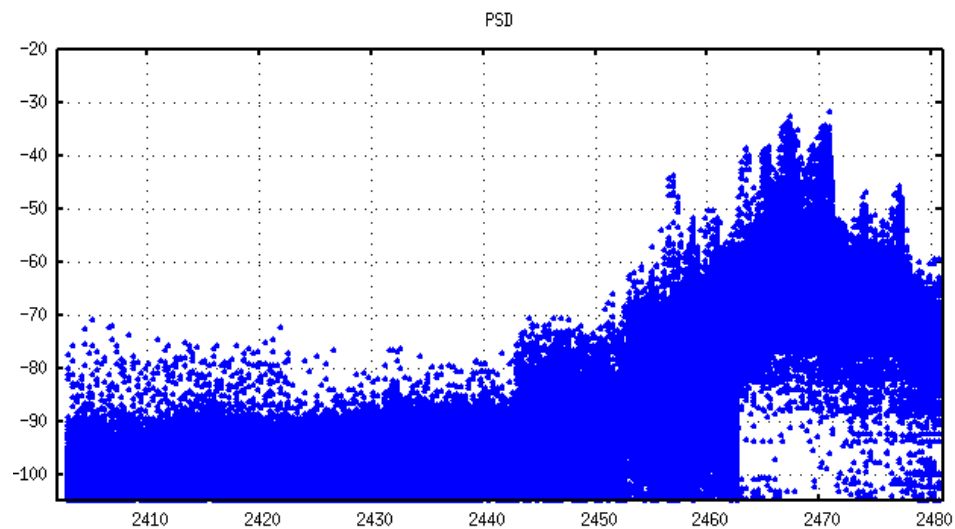


Figure 10: Atheros PSD graph

### ***Intel Wireless Adapters***

Besides Atheros Spectral Scan, a tool for retrieving channel information is available for Intel wireless adapters too. The CSI Tool is built on the Intel Wi-Fi Wireless Link 5300 802.11n MIMO radios, using a custom modified firmware and open source Linux wireless drivers.

The IWL5300 provides 802.11n channel state information in a format that reports the channel matrices for 30 subcarrier groups, which is about one group for every 2 subcarriers at 20 MHz or one in 4 at 40 MHz. Each channel matrix entry is a complex number, with signed 8-bit resolution each for the real and imaginary parts. It specifies the gain and phase of the signal path between a single transmit-receive antenna pair [10].

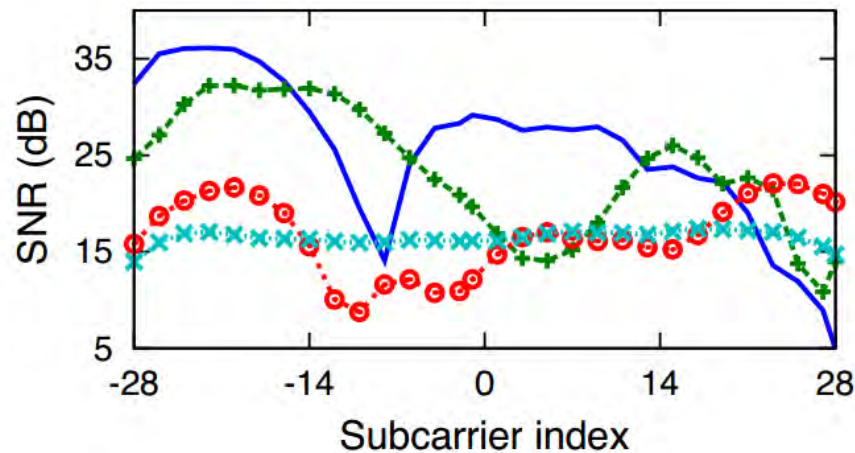


Figure 11: CSI tool

### ***Cisco Spectral Analysis***

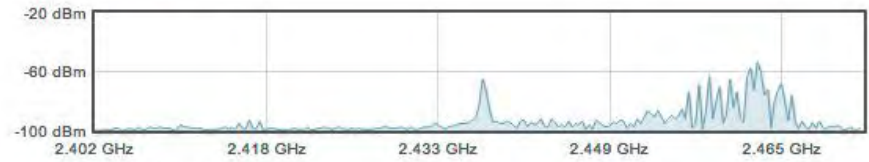
Spectral Analysis is a monitor tool offered by Cisco Meraki APs. The APs scan for both 802.11 (other APs) and non-802.11 sources of RF interference (eg. Bluetooth headsets, cordless phones and microwaves). All Meraki APs also scan for interference data, which is then fed into the Meraki Auto RF planning algorithms to determine optimal channel plan (if auto-channel selection is enabled) and transmit power settings. Real-time channel utilization scans can be run from the Live Tools section of the Access Point Details page, giving an administrator both instantaneous and historical data about interference sources in the area of a particular AP. The channel utilization live tool will provide interference information on the 2.4 GHz and 5 GHz. This framework gives user the ability to perform 2 types of network scans:

- Opportunistic scans, which are performed when an individual AP has no clients associated to it
- Mandatory scans, which are performed at a user-defined time of day (on specific days of the week) by all APs in the network [11]

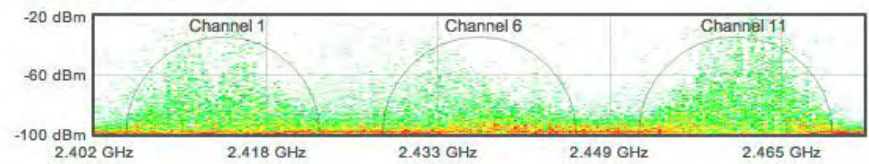
## Live tools

Current clients
Channel utilization
Spectrum analysis
Ping
Traceroute
Throughput
Blink LEDs
Reboot AP

### Spectrum analysis



### Cumulative distribution



RF overview > 3 C4 - using channels 1, 161

2.4 GHz 5 GHz

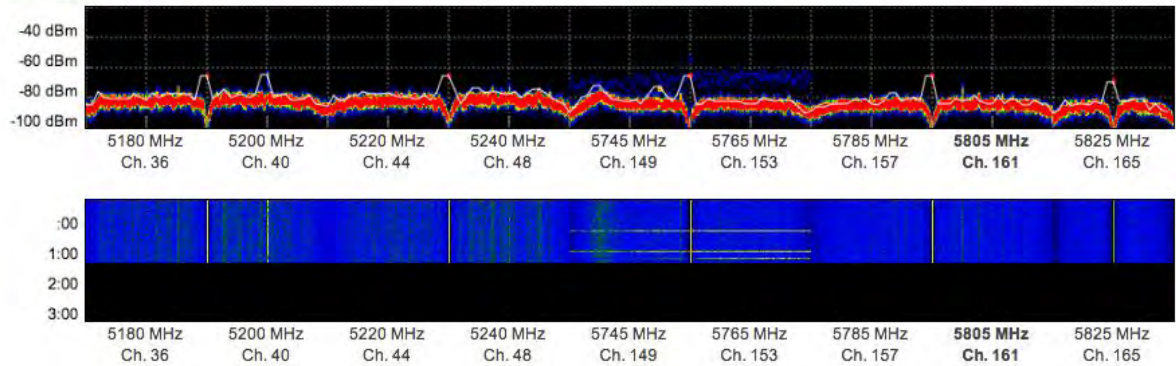


Figure 12: Cisco Spectral analysis visualization tools



## Chapter IV: Algorithm and Implementation

For this master thesis an algorithm for estimating the transmission opportunities of a commercial wireless adapter was developed. The commercial solutions which provide spectral information were extensively studied. Finally the algorithm implemented via Atheros chipsets and the backports open source driver. As referred in Chapter III this device provides spectrum information based on Energy Detection sensing. This gives the advantage of detecting both IEEE 802.11 and non IEEE 802.11 devices. Energy detection is able to map any form of transmission in a specified frequency range.

Atheros wireless chipsets have been used in the past for detecting non IEEE 802.11 devices[12]. The Airshark framework is able to detect non IEEE 802.11 using Atheros Spectral Scan tool. A fusion of Feature Detection using Energy Detection measurements is made on the aforementioned work. A weak aspect of this framework is the detection and feature extraction only for non IEEE 802.11 devices. The algorithm proposed by this master thesis, aim in detecting both IEEE 802.11 and non transmissions.

### ***Default Spectral Scan Implementation in Ath9k Driver***

This chapter contains at first the basic architecture of Spectral Scan. Then the approach used for this master thesis is being analyzed.

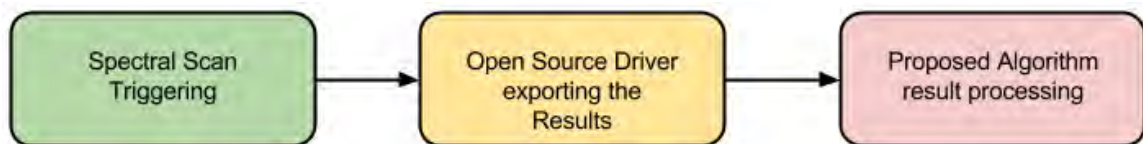


Figure 13: Spectral scan procedure overview

For performing a spectral scan the sequence shown in Figure 13 is used. First of all a bash script was developed. In this script the user can initialize some parameters which specify the Spectral Scan Results. For example

```
iw dev wlan0 scan
```

command triggers the driver for scanning all the center frequencies defined in the /net/mac80211/scan.c file.

```
iw dev wlan0 scan freq 2417 2422
```

The above command triggers the driver for performing a Spectral Scan only at the “2417” and “2422” center frequencies. In this script we can also find the paths in which spectral results are stored and the type of scan the user ask for the driver to perform. This script is consisted of typical linux console commands.

After the user executes the trigger script it is turn for the driver to perform the Spectral Scan core functionality. At this stage the driver suspends the programmed functions and trying to gather the spectrum information requested from the user. At the end of this process the `/sys/kernel/debug/ieee80211/phy0/ath9k/spectral_scan0` proc file is filled with the spectral measurements. The fact that the Spectral Scan operation uses open source driver, gives experimenter the opportunity of changing core driver parameters. In this chapter and in the section of “Implementation” we name briefly the changes have been made in the open source driver.

After driver finishes the Spectral Scan, the data are stored in the proc files for further analysis. An algorithm has been developed in order to make the processing of these data. The functions of this script are listed in the next section.

### ***Proposed Algorithm***

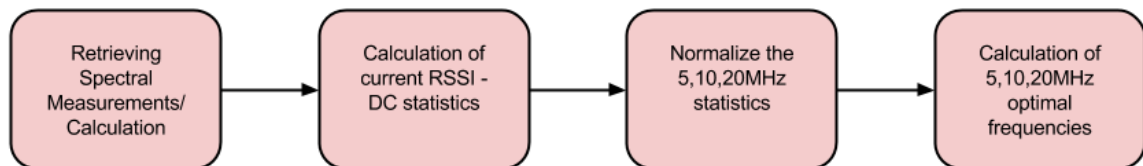


Figure 14: Proposed algorithm

### **Retrieving Spectral Measurements**

The exported data from the wireless card are stored in the proc files as a binary sequence. In this form it is not possible to take the desired spectrum information from them. The first part of the algorithm undertakes to store the binary data in appropriate form using structures. The produced structures contain full spectral information.

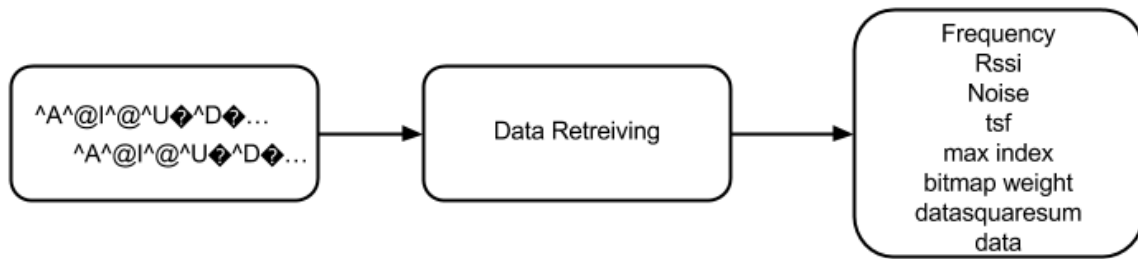


Figure 15: Exported data

In this work we mainly used frequency, rssi (received signal strength indicator) and tsf data.

### Calculation of RSSI Statistics

Now the data are stored on the structures and almost ready for analysis. The Atheros wireless adapters return 56BIN FFT snapshots in a 20 MHz channel. These samples need to be converted, in terms of being easily understood. This conversion also helps the experimenter to depict each FFT BIN in a PSD format. The formula for converting them is:

$$P_{dBm} = \text{noise} + \text{rssi} + 20 * \log_{10}(\text{DATA}) - \log_{10}(\text{datasquaresum}) * 10;$$

The Atheros Spectral Scan returns not one but several snapshots of 56BINS for each center frequency. In the channel of 20 MHz the card generates 56BINS, so there will be more than one BIN per MHz. The default number for snapshots, defined by the driver, is 8 and the maximum is 32. Through some modifications on the driver we managed to change the snapshots number from 32 to 255. This gives us the advantage of having better accuracy on the PSD graphs and more information for processing. On the “Implementation” section there will be more information about the changes made on the driver source code.

The number of channels for scanning is from 1 to 32, corresponding to the number of available channels in the entire 2.4 GHz and 5 GHz bands. Let’s assume that the user requests 15 channels for scanning and the result set number is 250. At the end of the above retrieving-converting procedure,

$$\text{Number Of Generated Samples} = \frac{\text{Num\_of\_BINS} * \text{Num\_of\_result\_sets} * \text{Num\_of\_channels\_for\_scanning}}{\text{Num\_of\_channels\_for\_scanning}}$$

$$\text{Number Of Generated Samples} = 56 * 250 * 15 = \mathbf{210000}$$

results will be produced. All results are stored into matrices and have the above format.

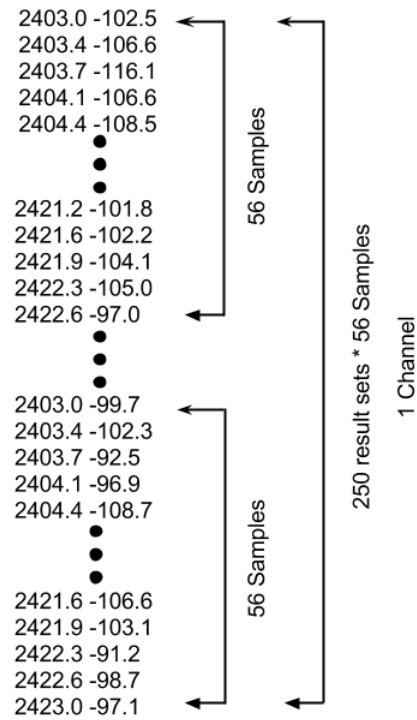


Figure 16: Number Of Generated Samples

Looking at each BIN separately cannot reveal useful information about the spectrum conditions. The proposed algorithm averages the data shown in Figure 16. For example 3 samples are given for the frequency “2403” as depicted in the Figure 16. The result for this frequency will be  $(-102.5 -106.6 - 116.1) / 3 = \mathbf{-108.4}$ . For each “integer” frequency the general formula of Figure 17 is applied. Where  $N_s$  is the number of generated samples per frequency( 2403, 2404, 2405 ..... ) and  $rss_i$  denotes the rssi value per sample. This procedure is repeated for the number of result sets requested from the user. Usually the difference between the samples of the same frequency is not so big, thus valuable information will not be easily wasted. Then the table of Figure 17 is further horizontally averaged.





capability of 40 MHz and 80 MHz respectively. The base IEEE 802.11 standard gives the opportunity for functioning in 5 MHz and 10 MHz too. Taking advantage of this, we decided to implement different DC evaluations for different bandwidth configurations.

Starting for the first central frequency  $f_c$ , we calculate the DC and RSSI (+ - bw/2 where bw is 5, 10, and 20). Then a sliding window loop is executed in order to calculate the rest data as depicted in Figure 19.

Frequency	5MHz DC	10MHz DC	20MHz DC	5MHz Rssi	10MHz Rssi	20MHz Rssi
2413	5.1	6.5	4.2	-68.3	-68.3	-68.0
2414	9.9	11.0	5.9	-67.5	-67.8	-67.0
2415	19.9	15.2	6.4	-63.2	-63.1	-63.6
2416	24.2	21.0	8.2	-61.3	-62.6	-61.5
•	•					•
•	•					•
•	•					•
2473	0.0	0.0	0.0	-97.3	-98.8	-97.1

Figure 19: Results matrix

### Normalize the Channel Statistics Over Time

Each spectrum scan responds to 20ms. The spectrum conditions are extremely unstable and concrete results need to be exported. Due to this the data captured in a Spectral Scan are normalized with data captured on previous scans. Specifically

$$\text{FinalResult} = 70\% \text{ of Current Result} + 30\% \text{ of Old Results}$$

### Calculation of Optimal Frequencies

At the end and under serial comparison of normalized data, the algorithm outputs the optimal frequencies of 5,10 and 20 MHz. The selection criterion of these is lowest DC values for each bandwidth as shown in the Figure 20.

2413	0.0	0.0	0.0	-102.4	-102.5	-102.6
2414	0.0	0.0	0.0	-102.4	-102.6	-102.6
2415	0.0	0.0	0.0	-102.4	-102.6	-102.7
2416	0.0	0.0	0.0	-102.6	-102.6	-102.8
2417	0.0	0.0	0.0	-102.6	-102.6	-102.8
2418	0.0	0.0	0.0	-102.8	-102.7	-102.9
2419	0.0	0.0	0.0	-103.0	-102.8	-103.0
2420	0.0	0.0	0.0	-103.0	-103.0	-102.9
2421	0.0	0.0	0.0	-103.1	-103.1	-102.7
2422	0.0	0.0	0.0	-103.2	-103.3	-102.6
2423	0.0	0.0	0.0	-103.3	-103.3	-102.6
2424	0.0	0.0	0.0	-103.4	-103.3	-102.6
2425	0.0	0.0	0.0	-103.4	-103.3	-102.6
2426	0.0	0.0	0.0	-103.5	-102.9	-102.6
2427	0.3	0.0	0.0	-70.0	-102.6	-102.7
2428	0.3	0.0	0.0	-69.3	-102.6	-102.6
2429	0.0	0.0	0.0	-102.1	-102.6	-102.6
2430	0.0	0.0	0.0	-102.1	-102.4	-102.6
2431	0.0	0.0	0.0	-101.9	-102.2	-102.6
2432	0.0	0.0	0.0	-101.7	-102.2	-102.5
2433	0.0	0.0	0.0	-101.5	-102.0	-102.2
2434	0.0	0.0	0.0	-101.6	-101.9	-101.8
2435	0.0	0.0	0.0	-102.1	-101.9	-101.4
2436	0.0	0.0	0.0	-102.5	-101.9	-101.1
2437	0.0	0.0	0.0	-102.4	-102.3	-100.7
2438	0.0	0.0	0.0	-102.4	-101.9	-100.3
2439	0.0	0.0	0.0	-102.5	-101.2	-99.6
2440	0.0	0.0	0.0	-101.6	-100.5	-99.1
2441	0.0	0.0	0.0	-100.5	-99.9	-98.7
2442	0.0	0.0	0.0	-99.5	-99.5	-98.5
2443	0.0	0.0	0.0	-98.5	-98.7	-97.4
2444	0.0	0.0	0.0	-97.7	-97.6	-96.2
2445	0.0	0.0	0.0	-96.5	-96.7	-95.0
2446	0.0	0.0	0.0	-94.7	-95.9	-93.7
2447	0.0	0.0	0.0	-94.3	-95.2	-92.4
2448	0.0	0.0	0.0	-94.1	-92.8	-90.9
2449	0.1	0.0	0.1	-94.0	-91.2	-89.4
2450	0.1	0.1	0.1	-91.3	-89.5	-87.8
2451	0.2	0.2	0.1	-88.5	-87.6	-86.1
2452	0.2	0.2	0.2	-86.0	-85.8	-84.2
2453	0.4	0.2	0.2	-83.8	-83.4	-82.3
2454	0.7	0.3	0.4	-81.4	-81.2	-80.5
2455	2.2	1.2	0.8	-78.1	-79.2	-78.8
2456	7.3	4.0	2.8	-68.7	-76.9	-69.5
2457	15.1	12.4	7.9	-68.9	-69.3	-69.2
2458	31.9	39.0	19.3	-68.8	-68.6	-68.8
2459	53.4	62.6	37.1	-67.9	-67.9	-68.7
2460	76.5	81.1	65.3	-66.9	-66.9	-68.5
2461	88.8	90.8	84.2	-65.1	-65.2	-67.7
2462	92.3	95.7	92.5	-63.5	-63.2	-66.6
2463	95.2	97.6	96.7	-61.2	-61.6	-65.2
2464	97.8	98.6	97.8	-58.8	-60.8	-64.7
2465	98.9	99.0	97.5	-57.5	-60.0	-64.2
2466	99.5	99.1	97.8	-57.0	-59.6	-63.9
2467	99.0	99.0	97.3	-57.1	-59.4	-63.9
2468	97.3	97.1	94.5	-57.8	-59.1	-63.9
2469	91.0	90.8	87.3	-58.6	-59.4	-64.1
2470	70.0	70.0	66.4	-59.9	-60.2	-64.3
Optimal Frequencies:						
5 MHz: 2426						
10 MHz: 2424						
20 MHz: 2419						

Figure 20: Optimal channels selection



## ***Implementation***

### **Algorithm**

The script for the proposed algorithm developed exclusively with C programming language. Use of basic structures and elements was made. For executing this script, there is no need of using high end devices. The script normally executed on devices with Atom processors.

### **Driver Modifications**

Taking advantage that the framework is running on a open source driver, changes was made in order to maximize the performance of the system.

- **Channelization:** The IEEE 802.11 standards propose 5 MHz space between center frequencies of 2.4GHz band and 20 MHz on the 5GHz band. Through the backports open source driver, a smart channelization was implemented. Particularly the space between the channels became 1 MHz. This provides the opportunity of tuning our wireless in many more center frequencies. On the standard channelization 14 channels offered on the 2.4GHz. With the proposed improvement 72 channels become available. Obviously the fact that the 2.4GHz band is highly congested, doesn't change only with better channelization. This feature still offers much more opportunities for avoiding interference from other users and devices.
- **Number of Scanned Channels:** As mentioned above, the spectral scan is an extra overhead on the adapter's scheduled functions. If someone uses the default driver, a typical scan of the 2.4GHz consists of 14 scanned channels. After extensive experiments observed that the information offered from this scheme is highly repeated and it is not fully necessary. The proposed method is to scan fewer channels for reducing the offline adapter's time. Figure 24 contains scan information from 7 channels. As it is obvious 7 channel scans are able to produce sufficient spectrum information.
- **Number of Result Sets per Channel:** The backports driver returns by default 8 result sets per channel. The maximum number of results set can be up to 32. By changing several parameters in the source code, this number increased to 250 per channel.

## Graphs

Gnuplot utility [16] was used in order to generate the graphs for the RSSI and Duty Cycle metrics. Gnuplot is a portable command-line driven graphing utility for Linux, OS/2, MS Windows, OSX, VMS, and many other platforms. It is a powerful tool which used not only for static representation of the measured metrics, but it is suitable for real time plotting too. We managed to develop a real time plotter with refresh rate of 0.5sec.

## Chapter V: Experiments

This chapter aims to present and analyze the results was obtained, by using the proposed algorithm in real environments. Our proposed algorithm detects the congestion created by several devices in this section. We want to measure the Duty Cycle of each tested device and export some general features of them. The algorithm is able to detect both IEEE 802.11 and non devices.

### **IEEE 802.11 Devices**

#### **5GHz IEEE 802.11 Interferer**

The first experimental setup is as follows:

- Our link is transmitting on the 5GHz spectrum and specifically on the 5650 MHz frequency. This links sends data with 100Mbps.
- Our interferer will transmit in the exact same frequency as the primary link. Gradually will start sending data from 10Mbps to 100Mbps.

This scenario is intended to show the gradual reduction on the primary's link performance in comparison with the increasing of the interference link's Duty Cycle. Below in the Table 1the experiment results are detailed.

Interferer (Achieved / Demanded)	Primary Link (Achieved / Demanded)
10 Mbps / 10 Mbps	99.3 Mbps / 100 Mbps
20 Mbps / 20 Mbps	92.5 Mbps / 100 Mbps
29.8 Mbps / 30 Mbps	84.1 Mbps / 100 Mbps
37.6 Mbps / 40 Mbps	73.2 Mbps / 100 Mbps
45.5 Mbps / 50 Mbps	71.2 Mbps / 100 Mbps
46.2 Mbps / 60 Mbps	69.9 Mbps / 100 Mbps
47.7 Mbps / 70 Mbps	68.1 Mbps / 100 Mbps
47.9 Mbps / 80 Mbps	68.6 Mbps / 100 Mbps
48.2 Mbps / 90 Mbps	68.9 Mbps / 100 Mbps
49.1 Mbps / 100 Mbps	70.1 Mbps / 100 Mbps

Table 1: Scenario Results

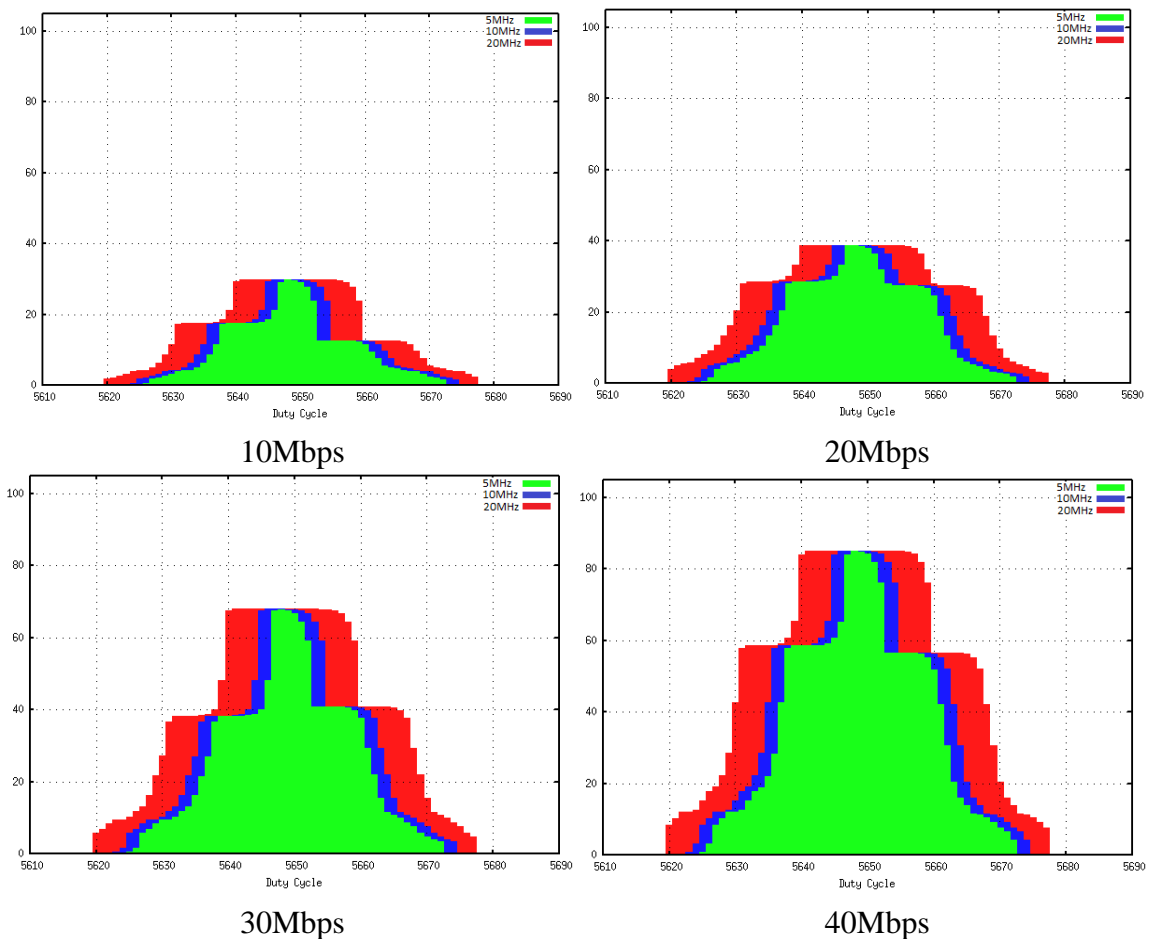
In the Figure 21 the progressive increase of interferer's Duty Cycle can be observed. Figure 22 depicts the power spectral density incensement in comparison with the interferer's throughput demand.

Interferer Demand	5 MHz DC	10 MHz DC	20 MHz DC
10 Mbps	28.6	29.0	29.1

20 Mbps	37.9	38.6	38.6
30 Mbps	66.7	67.8	67.9
40 Mbps	84.2	85.0	85.1
50 Mbps	85.7	86.5	86.6
60 Mbps	86.3	87.2	87.3
70 Mbps	90.4	91.1	91.1
80 Mbps	95.1	95.5	95.5
90 Mbps	95.3	95.9	95.3
100 Mbps	96.0	96.5	96.6

Table 2: Duty Cycle Results

Table 2 lists the percentage of Duty Cycle depending on the interferer's demand. It can be observed that primary's link performance decreases gradually. When the demand of the interferer's link is between 40 and 100 Mbps we don't notice remarkable difference on the primary's link performance. This can be confirmed both with Table 1 and Figure 21.



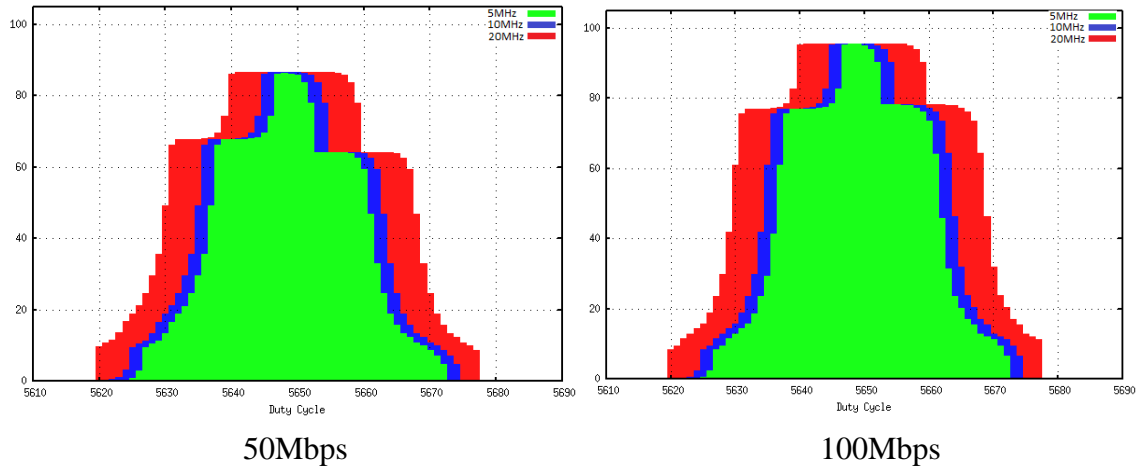


Figure 21: Interferer Duty Cycle

As seen in Figure’s 21 subfigures, we can also note that the Duty Cycle for the 20 MHz channels spreads further than the others. It makes perfect sense, since the observed frequency range is larger than the 5 MHz and 10 MHz channels.

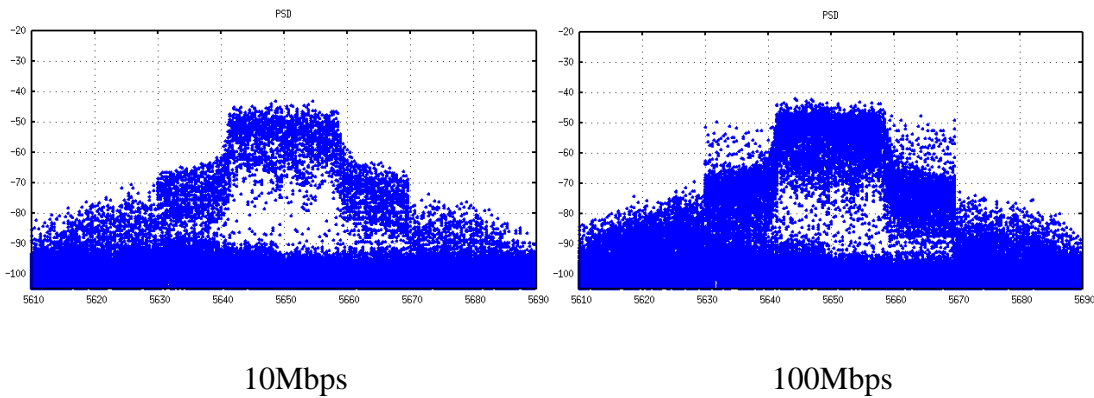


Figure 22: Interferer RSSI

Comparing the Figure 21 with Figure 22 we can understand how important the Duty Cycle calculation is. Even though in the Figure 22 it is obvious that the spectral density becomes thicker, we cannot calculate the spectrum occupancy only with the RSSI metric.

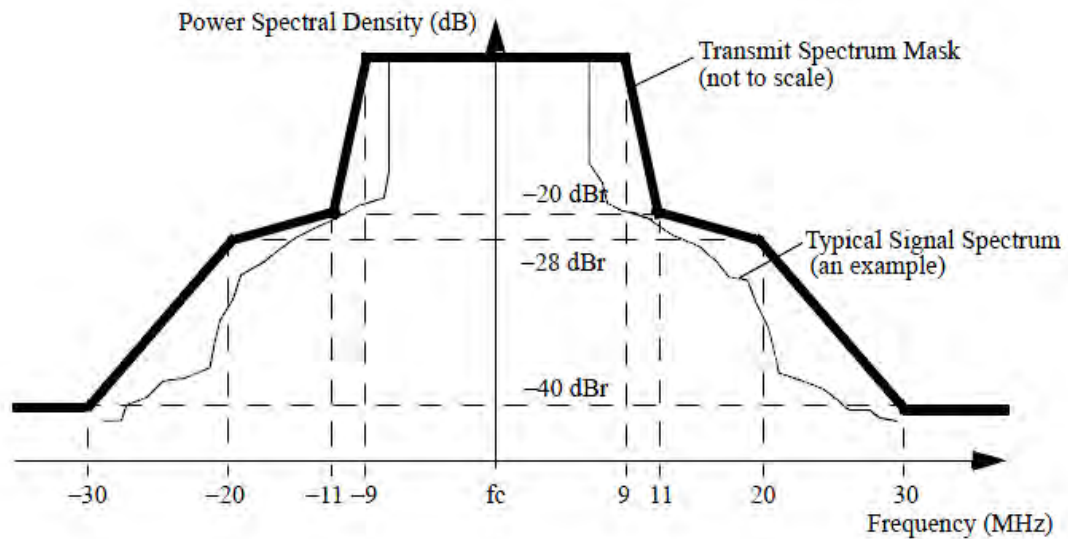


Figure 23: IEEE 802.11 transmit mask[19]

Comparing the Figure 22 with Figure 23, we can note that the wireless card reports data tending to the theoretical transit mask of the IEEE 802.11 standard.

### ***Non IEEE 802.11 Devices***

The following section analyzes the interference caused by non IEEE 802.11 devices.

#### **Microwave Oven**

Microwave ovens function at the 2.4GHz ISM band too. These devices have approximately 50% theoretical Duty Cycle. In this scenario a microwave oven was placed about 5 meters away from the sensing device. The spectrum was captured when the microwave was active and the results are depicted in the Figure 24.

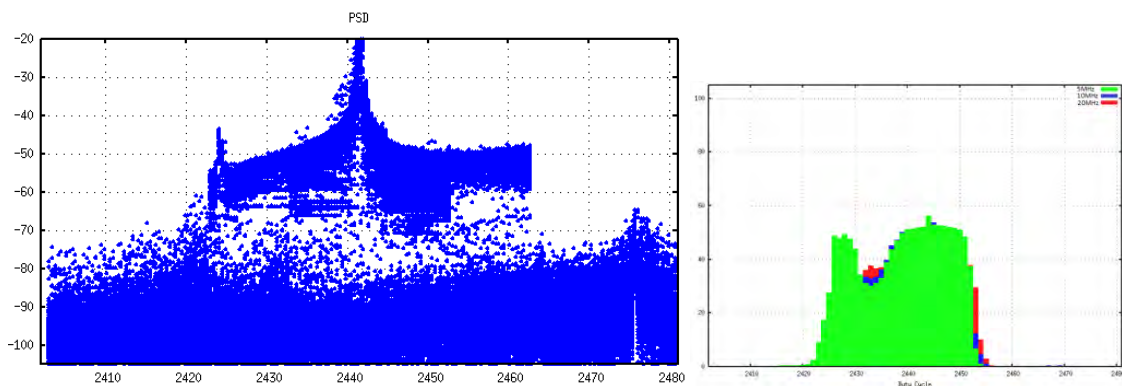


Figure 24: Microwave Oven RSSI and Duty Cycle

Observations exported of this scenario:

- From Figure 24 we can note that the transmit power of the microwave oven is larger than a common IEEE 802.11 and its value is about -40dBm
- The frequency range that this device functions is 2420 – 2460. We executed the experiment many times and observed that the frequency range changes a little bit every time.
- The detected Duty Cycle is indeed about 50% in the central frequency region.

This device pollutes a large proportion of the spectrum. The best channels for performing a transmission when microwave oven is active seem to be 1 and 11.

## Quadcopter Radio Controller

In this scenario a common quadcopter R/C device was tested. This device performs as a frequency hopper and its transmit power is about 500mW. This device also placed about 5 meters from the sensing device.

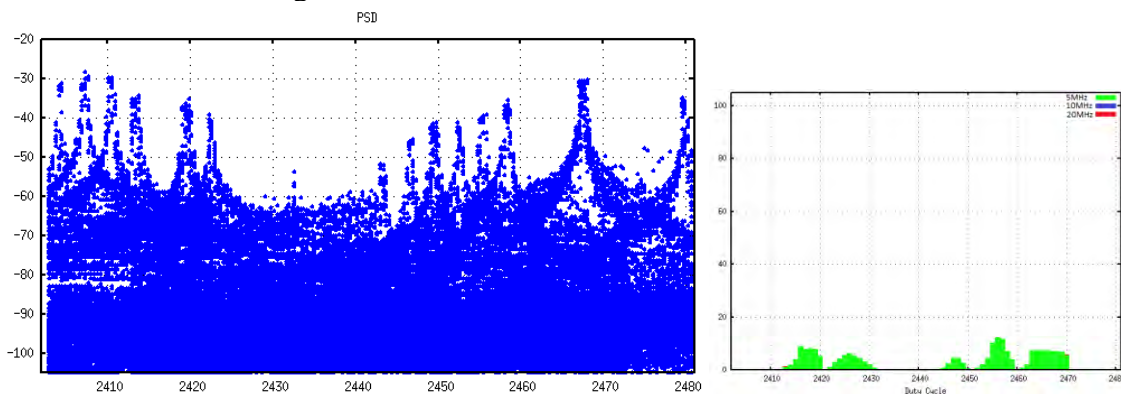


Figure 25: Quadcopter R/C RSSI and Duty Cycle

Observations exported of this scenario:

- Quadcopter R/C is identified as a heavy interferer, due to its relatively high transmission power that is shown to exceed the 200 mW limit of commercial 802.11 devices
- Its transmissions are narrowband, so an IEEE 802.11 if working on 20 MHz bandwidth wont back-off. This is because the  $P_{TH}$  will not exceed -80 dBm.

## Bluetooth

A file exchange between two Bluetooth v4.0 devices was made, in order to capture the transmissions on the 2.4GHz band. This device performs as a frequency hopper too and its transmit power is about 100mW.

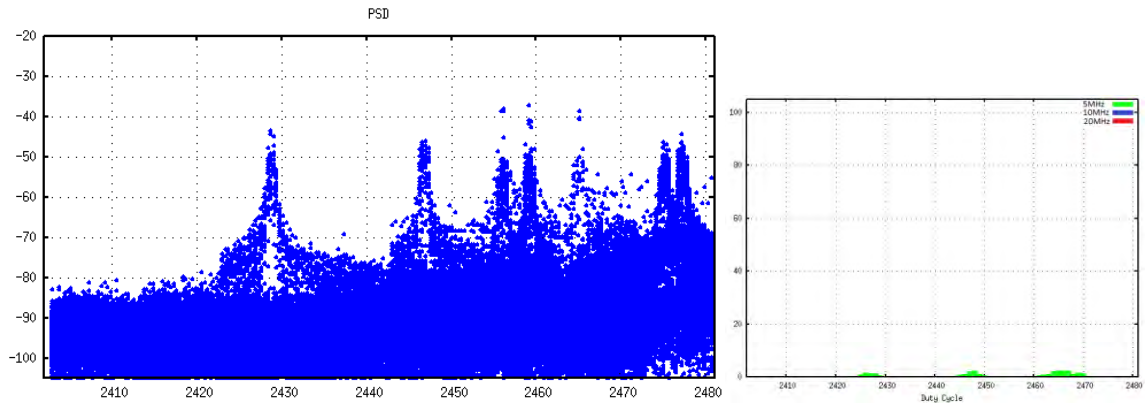


Figure 26: Bluetooth RSSI and Duty Cycle

Observations exported of this scenario:

- This device performs a frequency hopping on specific frequencies, which responds to the Adaptive Frequency Hopping capability of bluetooth
- It is a narrow band transmission and affects minimally the 5 MHz channels as shown on the Duty Cycle graph

## Wireless Camera

Most of the commercial wireless camera are using the 2.4GHz band too. The theoretical bandwidth is 18 MHz and its transmission power 10 dBm. You can set this device on 4 channels. We fixed one with central frequency of 2432 MHz.

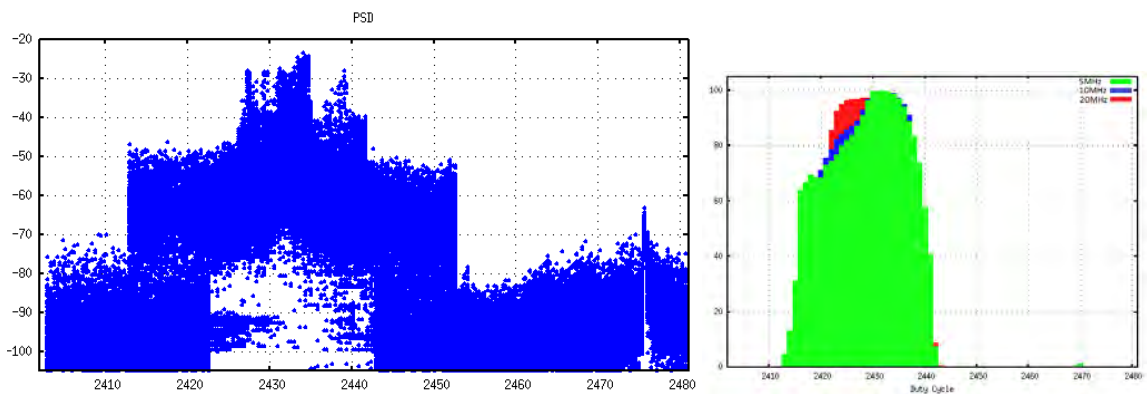


Figure 27: Wireless Camera RSSI and Duty Cycle

Observations exported of this scenario:

- The emissions detected with a high signal strength of -30dBm.
- The frequency range that this device functions is 2415 – 2455. Based on the manufacture's manual our device bandwidth is 18 MHz. In practice we see the phenomenon of power leaking intensively.
- The detected Duty Cycle is about 100% in central frequency range



## Chapter VI: Conclusions

In this master thesis an overview of the Cognitive Radio Networks was presented. Great importance was given on implementing these networks on commercial devices. We tried to show how the generalized method of Energy Detection in combination with the Atheros wireless adapters, can export useful information for the spectrum conditions. Then an algorithm for monitoring the spectrum and choosing the optimal channels was designed and implemented. Finally several experiment with different architectures were executed.

Taking into consideration the results of the above experiments and the general behavior of the Atheros wireless adapters during the spectrum sensing procedure, it is clear that these devices are able to provide spectrum information respective with high end devices.

The proposed algorithm analyzed in this master thesis, is a part of our published work [17][18].

## References

- [1] Amir Ghasemi, Elvino S. Sousa “Spectrum Sensing in Cognitive Radio Networks: Requirements, Challenges and Design Trade-offs”
- [2] Wei Wang “A brief survey on cognitive radio”
- [3] Tefvik Yucek, Huseyin Arslan “A survey of spectrum sensing algorithms for cognitive radio applications”
- [4] Ian F. Akyildiz et al. “Next generation/dynamic spectrum access/cognitive radio wireless networks: A survey”
- [5] Danijela Cabric et al. “Implementation issues in spectrum sensing for cognitive radios”
- [6] Hüseyin Arslan “Cognitive Radio, Software Defined Radio, and Adaptive Wireless Communication Systems”
- [7] National Instruments <http://www.ni.com/white-paper/14311/en/>
- [8] Ettus Research / National Instruments  
“[https://www.ettus.com/content/files/07495\\_Ettus\\_N200-10\\_DS\\_Flyer\\_HR\\_1.pdf](https://www.ettus.com/content/files/07495_Ettus_N200-10_DS_Flyer_HR_1.pdf)”
- [9] Ath9k spectral scan “[http://wireless.kernel.org/en/users/Drivers/ath9k/spectral\\_scan/](http://wireless.kernel.org/en/users/Drivers/ath9k/spectral_scan/)”
- [10] 802.11 CSI tool “<http://dhalperi.github.io/linux-80211n-csitool/>”
- [11] Cisco Spectral Analysis “<https://docs.meraki.com/display/MR/Spectrum+Analysis>”
- [12] Shravan Rayanchu, Ashish Patro, Suman Banerjee “Airshark: Detecting Non-WiFi RF Devices using Commodity WiFi Hardware”
- [13] Simon Wunderlich “[https://github.com/simonwunderlich/FFT\\_eval](https://github.com/simonwunderlich/FFT_eval)”
- [14] Mathworks “<http://www.mathworks.com/>”
- [15] IEEE 802.11-2007 Wireless LAN Medium Access Control and Physical Layers Specifications., 2007
- [16] GNU Plot tool “<http://www.gnuplot.info/>”
- [17] V. Passas, K. Chounos, S. Keranidis, W. Liu, L. Hollevoet, T. Korakis, I. Koutsopoulos, I. Moerman, L. Tassiulas, "Online Evaluation of Sensing Characteristics for Radio Platforms in the CREW Federated Testbed"
- [18] Stratos Keranidis, Kostas Chounos, Thanasis Korakis, Jordanis Koutsopoulos and Leandros Tassiulas “Enabling AGILE Spectrum Adaptation in commercial 802.11 WLAN deployments”
- [19] Transmit mask “<http://m.policytracker.com/free-content/blogs/pierre-de-vries/is-2.4ghz-wi-fi-the-next-gps-lightsquared>”