Master's Thesis

Location collector platform for indoor tracking

By

Ricardo Morales González and Miguel Sanz Rodríguez

Department of Electrical and Information Technology
Faculty of Engineering, LTH, Lund University
SE-221 00 Lund, Sweden

Abstract

Human mobility patterns have become of great importance in different topics like urban design as one of many examples in indoor environments. The understanding of how human moves within an area remain limited due to the lack of tools for monitoring the timing and location of individuals and the necessity of user intervention.

Qubulus AB is an example of provider of an application position platform for indoor tracking called LocLizard. LocLizard is based on a client application installed on a user mobile phone. It collects radio downlink signals to create a map with fingerprints that is used to calculate the position of the mobile device. The analytics tool gives you the capability to analyze historical data to see mobility patterns. However, the historical data capable to be recorded is limited to the number of mobile devices, which have Qubulus client software [1]. Other providers with similar solutions are e.g. Polestar and Insiteo.

Our thesis project is to find a way to increase the number of mobile devices that can be positioned within an indoor environment without a client application installed on the mobile device. Furthermore, in order to reduce user intervention, it is important to know if we can estimate the position of the mobile device when it is in the idle mode state, i.e. when the mobile device interacts with its backbone network on a M2M basis.

Our approach is to collect uplink signals from the Global System for Mobile Communications (GSM), Universal Mobile Telecommunications System (UMTS), and Wireless Local Area Networks (WLANs) that can be used for positioning and tracking based upon location method, such as trilateration or fingerprinting.

Three questions are to be answered: How often does the mobile device send signals during the idle mode state for the different radio sources? Are these uplink signals feasible for positioning and tracking? Is there a way to find uniqueness between different mobile devices (terminals)?

Through our analysis and experimental results, trilateration as a location method reports a positive performance.

Acknowledgments

There are many people who we would like to thank for their encouragement and support in making our period of study a pleasant time. Here we can only mention a few of them.

We would like to thank our supervisor, Henrik Hedlund, for his valuable guidance in this master thesis.

We would like to thank the department of Electrical Information Technology (EIT) of Lund Tekniska Högskola (LTH) and Professor Anders Johansson for their helpful advice during this project.

We would like to thank Anders Berkeman for his wise advice to some key aspects in this master thesis.

We would like to thank the department of Mathematical Statistics for their contribution to this research.

And of course, thanks to our family and friends for their support during our research within this master thesis.

Ricardo Morales González and Miguel Sanz Rodríguez

Table of Contents

CHAPTER 1

1 Introduction

1.1 Scope

The necessity of knowing the location of people is increasing day by day; at the same time, technology continues revolutionizing our world. Location Based Services (LBS) have acquired a huge importance for both mobile operators and users, since they posses great commercial opportunities but also provide great utility for users in daily life. The GPS (Global Positioning System), that a few years ago came up as an innovative technology, is nowadays a widespread service. GPS technology has proven successful technology for outdoor environments, but no technology has yet gained a similar wide-scale adoption for indoor positioning. This is a new exciting research scope that has just started to develop, but is evolving quickly as the demand of indoor localization increases.

There are plenty of uses of indoor localization. Airports, malls, and property owners are just a few examples of organizations that can get benefits using indoor positioning. For instance, the airports are nowadays looking for ways to raise customer satisfaction in a very competitive environment. To give the traveler the possibility to position themselves accurately opens up for services where they can get accurate calculations on how long it will take them to get to the gate combined with immediate information on departure & check in time, potential delays and much more. For malls and property owners' indoor localization brings them the technical possibilities to meet their customers all the way from approaching their location, entering and moving around indoor. Combining store and product search with event marketing, discrete advertising, offers and coupons, all controlled by the position of the customers, means definitely a huge commercial potential [2].

There exist many different positioning methods that given suitable measurements can be used to estimate sensor positions. In recent years, positioning systems for indoor areas using the existing wireless networks such as GSM, UMTS and local area network (WLAN) infrastructure have

been developed. Remarkable techniques are angle-of-arrival (AOA), time-of-arrival (TOA), multilateration or location fingerprinting. Fingerprinting method uses the received signal strength (RSS) that the mobile device receives from wireless networks at the sampling locations to build a "radio map" for the target environment. Once the radio map is built, the position is computed by comparing a new fingerprint with the ones forming the radio map, using different algorithms. Location fingerprinting performs well for non-line-of-sight (NLOS) circumstances and LOS environments are not required. It is generally agreed that a desirable indoor location system should be characterized by high accuracy, short training phase, cost-effectiveness (preferably using off-the-shelf hardware), and robustness in the face of previously unobserved conditions. Such schemes can provide value-added services for existing WLANs and 3G networks. Indoor localization techniques using location fingerprints are gaining popularity because of their cost-effectiveness compared to other infrastructure-based location systems. The fingerprinting technique is relatively simple to deploy compared to the other techniques such as angle-of-arrival (AOA) and time-of-arrival (TOA). Moreover, there is no specialized hardware required at the mobile station (MS). Any existing wireless LAN or 3G infrastructure can be reused for this positioning system [3][4].

Qubulus is one of the most promising indoor positioning start-ups, and it bases its technology in location fingerprinting. They use RSSI, from existing radio access networks such as GSM, UMTS and WLAN, at mobile devices to create the radio map that enables to compute the 3D position when a user requires so. This method gives ~ 3 meter accuracy in X and Y axis, and ~ 1 meter in Z axis. Currently the technology is based on a free client application installed in an android mobile device, in which users can see their current position [2].

There are other companies like GloPos, which has a similar technology collecting radio signals from different sources and applying some algorithms for the positioning of the mobile phone in an indoor environment.

1.2 Problem statement

The current techniques of indoor localization are based on signals received in the downlink (from Base Station or Access Point to the mobile device). For this reason, they require a client application installed on the mobile

devices, and therefore active user intervention is essential.

An increasing important demand of knowing mobility patterns of people in indoor scenarios has turned into a great commercial potential. For this purpose, to have client applications installed on every terminal and to need that the user intervenes actively are a clear obstacle. To develop a system that makes possible to detect and position mobile devices without any application installed on the terminals and without any (or minimum) active user intervention is the challenge that this master thesis faces.

1.3 Research approach

An alternative and novel approach for indoor positioning is developed in this master thesis. Instead of using RSS from downlink signals, we detect uplink GSM, UMTS and Wi-Fi signals coming out from mobile devices in order to get power levels that can be used for computing position using any indoor localization technique such as fingerprinting or trilateration. We focus our research in the idle mode of the three quoted technologies since is the most common scenario. Since this is a novel approach, the challenge is to prove the feasibility of uplink signals to be used for computing indoor position.

For the purpose of analyzing UMTS and GSM signals we combined Software defined radio (SDR) GNU Radio with Universal Software Radio Peripheral (USRP) equipment. These tools are used to design and implement a wireless spectrum sensor that captures traffic from mobile devices and store it in files that are further processed in Matlab.

In order to analyze Wi-Fi signals two approaches that differ on the way that the signals are detected are employed. Firstly, we utilized GNU Radio and USRP to detect and characterize uplink Wi-Fi signals. Secondly, LinkSys WRT54GL routers are used to sense the IEEE 802.11 spectrum and detect uplink packets, together with packet network analyzers such as Wireshark and tcpdump. The firmware DD-WRT, which is a Linux based alternative OpenSource firmware suitable for a great variety of WLAN routers and embedded systems, is installed on the routers in order to provide additional capabilities.

In all tests and measurements, two different mobile smartphones (HTC Desire Z and Samsung Wave GT-S8500) were used in order to analyze the

possible differences between vendors.

1.4 Related work

As this is a novel approach to indoor positioning, we have not found any related work similar to our master thesis. Obviously we have used plenty of bibliography and references from academic papers and dissertations that have been relevant to our research, but it is worth to mention that this master thesis presents a novel and innovative method for indoor localization that nobody has researched into before.

1.5 Thesis organization

The outline of the whole dissertation is presented below:

- Chapter 2 explains the theoretical background needed to understand this master thesis. It provides an overview of the three radio access technologies employed in the project and a detailed explanation of the idle mode in each of these technologies.

- Chapter 3 presents the analysis of UMTS and GSM uplink signals using GNU Radio and USRP equipment. The feasibility of these signals for the purpose of indoor positioning is discussed here.

- Chapter 4 describes the implementation of two different approaches for the purpose of detecting IEEE 802.11 uplink signals. For every approach, the method of detection, scenario and procedure of measurements are explained in detail.

- Chapter 5 provides experimental results of each approach with the corresponding explanations. For the LinkSys routers approach, an example of the use of the results to compute specific indoor positioning is presented, using trilateration technique.

- Chapter 6 summarizes the conclusions obtained after the work within this master thesis and future work.

CHAPTER 2

2 Theoretical background

2.1 Indoor positioning techniques

Indoor positioning has become an important research topic for industrial and academic purposes. Accurate localization in an indoor environment can enable localization aware services in many domains. Many domains can get benefits from indoor localization services. Furthermore, localization information of mobile units can be recorded to provide historical data of mobility patterns. Hence, the historical data recorded can be used for better designs, distribution of infrastructure among others regarding the domain.

There are many position techniques that have been proposed; however, not all techniques are suitable for indoor environments due to multipath effect. Fingerprinting is one that could overcome the multipath drawback.

2.1.1 Fingerprinting

Fingerprinting consists of two stages: A training phase where the signal strengths from different radio sources are collected (like GSM, UMTS, and Wi-Fi) at reference points and store together with their physical coordinates on a server; they are called fingerprints. In the positioning determination phase, the mobile device performs signal strength measurements and is continuously uploaded to the application back-end in real time, or at a configurable rate of frequency. The signal strengths collected are matched for similar patterns with different algorithm in the database. The closest match is selected and returned as the estimated position[2].

The signal strength is sensitive to the environment. Thus, a collection of access points can be chosen for fingerprinting location. The following factors can affect the signal strength [5].

1. Temperature: it affects the signal strength and can vary the fingerprints.
2. Sampling time: signal strength may differ from the same location in sampling time.
3. Granularity: The distance between two adjacent locations, where signals strengths are collected, affects the accuracy of location sensing. For our case, it was taken 5 meters away.
4. Pattern algorithm recognition: the most important part in location fingerprinting to compare multiple fingerprints stored in the server and picks the one that best matches the observed RSS.

2.1.2 Trilateration

The uplink data collected in this thesis project will be given to the Mathematical Statistics Department at Lund University where it will be analyzed to create algorithms for indoor positioning techniques.

Why trilateration? As an alternative of the fingerprinting technique, trilateration was used in a simple way to estimate the position.

Trilateration is the process to calculate the position of points by measurement distances. Trilateration has practical applications in surveying and navigation, including Global Positioning Systems (GPS).

Trilateration can be either on two or three dimensions. The difference is that we have two circles in two dimensions and spheres in three dimensions. When a point lies on two circles, the circle centers and the radii provide enough information to narrow the possibilities locations down to two. [6]

In three dimensional, when the point now lays on three circles, the centers of the circle as well their radii down the possibilities to no more than two. If it is increased the number of circles, it can be reduce the possibilities of locations.

The intersection of the surfaces for the three spheres is found by formulating the functions of them.

$$d1^2 = (x - x1)^2 + (y - y1)^2 + (z - z1)^2 \quad (1)$$

$$d2^2 = (x - x2)^2 + (y - y2)^2 + (z - z2)^2 \quad (2)$$

$$d3^2 = (x - x3)^2 + (y - y3)^2 + (z - z3)^2 \quad (3)$$

By resolving the equations (1), (2), and (3) it can be obtained x, y and z and get the location of the point.

2.2 Global System Mobile (GSM)

2.2.1 Overview

GSM is a digital wireless network standard designed by standardization committees from major European telecommunications operators and manufacturers. The GSM standard provides a common set of compatible services and capabilities to all mobile users across Europe and several million customers worldwide [7]. In this section we will describe the GSM features that are relevant to our research.

2.2.2 Air interface and multiple access

GSM employs a combined Frequency Division Multiple Access (FDMA)/Time Division Multiple Access (TDMA) approach which further combines with Frequency Division Duplex (FDD). In GSM 900, the most spread version of GSM, FDD is used to multiplex the uplink (890-915 MHz) and the downlink (935-960 MHz). Frequencies always exist in pairs, one for uplink (Mobile Station to Base Station) and one for downlink (Base Station to Mobile Station). The frequency spacing between the uplink and downlink for any given connection is 45 MHz. However, in the case of certain channels called “common channels” it is possible that the uplink and downlink frequencies serve different purposes and may not have any relation with each other. In contrast, channels that are dedicated for a Mobile Station (MS) are always allocated in pairs [8].

The air interface of GSM uses FDMA to operate over various frequencies separated 200 KHz. The outer 100 KHz of each 25 MHz band are not used, as they are guard bands to limit interference in the adjoined spectrum which

is used by other systems. The remaining 124 duplex channels are numbered by the so-called *Absolute Radio frequency Channel Numbers* (ARFCNs). In each of these sub bands, GSM performs TDMA to split the time axis in 8 time slots, which are periodically available to each of the possible 8 users. The collection of these 8 time slots repeated periodically over time is called a TDMA frame. The TDMA frames on the uplink are transmitted with a delay of 3 time slots with regard to the downlink. Each time slot that recurs in every TDMA frame and operated over a specific frequency is viewed as the `physical channel´ that carries user/signaling information. This physical channel lasts for 15/26 ms or 576.9 microseconds. The information is carried at a modulation rate of 270,833 Kbps and it is equivalent to 156.25 bits. Within each frame the timeslots are numbered from 0 to 7, and each subscriber accesses one specific timeslot in every frame on one frequency sub band. A MS uses the same time slots in the downlink as in the uplink, i.e. the time slots with the same number. Owing to the shift of three time slots, a MS does not have to send as the same time as it receives, and therefore does not need a duplex unit [8][9].

2.2.3 Modulation

The modulation used in GSM is Gaussian Minimum Shift Key (GMSK), which is a continuous-phase frequency-shift keying modulation scheme. It is similar to standard minimum-shift keying mobile (MSK); however the digital data stream is first shaped with a Gaussian filter before being applied to a frequency modulator. Special advantages of this modulation scheme are the narrow transmitter power spectrum with low adjacent channel interference and a constant amplitude envelope which allows the use of simple amplifiers in the transmitter with special linearity requirements [8].

2.2.4 Slow frequency hopping

Mobile radio channels suffer from frequency selective interferences, e.g. frequency selective fading due to multipath propagation phenomena. This selective frequency interference can increase with the distance from the base station, especially at the cell boundaries and under unfavorable conditions. Frequency hopping changes the transmission frequencies periodically and thus average the interference over the frequencies in one cell. This improves the Signal to Noise Ratio (SNR) to a high enough level for good speech quality [10].

GSM provides a slow frequency hopping which changes to a different frequency with each burst. The scheme is based on the idea that every mobile station transmits its TDMA frames according to a sequence of frequencies specified by the frequency hopping algorithm. A mobile station transmits on a fixed frequency over one time slot (577 microseconds) and then jumps to another frequency before the next TDMA frame. There are 64 frequency hopping sequences in GSM, each one containing up to 64 frequencies. The result hopping rate is about 217 changes per second, corresponding to the TDMA frame duration [10].

2.2.5 Uplink logical channels

In GSM the channels are divided into two groups: physical and logical channels. A physical channel corresponds to a specific timeslot on one carrier, while a logical channel reflects the specific type of information that the physical channel carries. The logical channels are mapped to the physical channels, and they carry different information depending on the type of logical channel.

The logical channels are divided into traffic and control channels. The traffic channels are the resources available for the users to send data or voice. The control channels are used for signaling and controlling the traffic channels [10]. Below the uplink logical channels are shortlisted [10]:

The Random Access Channel (RACH): this channel is used initially by the MS to attempt accessing the network. When the MS responds to a paging request or when initiating a call, the MS is requesting a signaling channel.

The Stand-alone Dedicated Control Channel (SDCCH): this bi-directional channel carries all the signaling information. It can be used for authentication, ciphering, call set-up or transmission of text messages.

The Slow Associated Control Channel (SACCH): this bi-directional channel is used to transfer signaling information during an ongoing call on a traffic channel. In the uplink the MS reports the downlink measurements to the BTS. It is used for non-urgent procedures, and can carry up to two messages per second in each direction.

The Fast Associated Control Channel (FACCH): this bi-directional channel is used when there is a need of higher capacity signaling in parallel with ongoing traffic, and it is mainly used for handover procedures.

2.2.6 Frame hierarchy

The physical content of each time slot is called a `burst´. There are different types of bursts, each of which has a different structure. Each burst maps to a time slot, and they form the lowest level in GSM lever hierarchy. The TDMA frame thus forms the next level in the hierarchy. At the next level there is a collection of TDMA frames called the `multi-frame´ that is of 2 types, 26-frame multi frame and 51-frame multiframe. The 26-frame multiframe is used for traffic channels wherein 24 channels are used for actual traffic exchange, while one is used for low-rate signaling that accompanies any traffic exchange, and one channel is left idle. In contrast, the 51-frame multiframe is used for signaling. The difference at the multi-frame level is removed at the `superframe´ level wherein a superframe always has 1326 frames (26x51 or 51x26). The last level of the hierarchy is the `hyperframe´ which comprises 2048 frames. Thus, the GSM hierarchy repeats itself after an interval of 3 hours 28 minutes 53 seconds 760 miliseconds [9].

2.2.7 GSM burst and their types

A burst lasts for 156.25 bit duration or a period of 576.9 microseconds. During this period, the signal reaches from amplitude of 0 to a specified value. Thereafter, the signal is modulated and transmitted. At the end of the transmission, the signal reaches back to 0. The transmitted signal here is viewed to be the modulated signal obtained after modulating a signal that comprises a series of 1´s followed by the information to be sent which is followed again by a series of 1´s. In order to ensure efficient demodulation, 3 tail bits (all 0´s) are added before and after the information contents to ensure the transition of the signal (from 1 to 0 and back from 0 to 1) during the transfer. The guard period is used to ramp up and down the signal [9]. There are different types of bursts sent in the uplink [9]:

Normal burst (NB): this burst is used to carry information on traffic and control channels, except for RACH (Random Access Channel). It contains

116 encrypted symbols and includes a guard time of 8.25 symbol duration (30.46 μs).

Access burst (AB): this burst is used in order to help the mobile to find its distance. Its useful contents are much shorter than ones of the normal bursts to allow for propagation delays8 only 36 information bits can be carried). When the mobile first attempts to access the network, it does not provide its own identity; instead, it seeks a channel on which it can send further information needed for authentication and resource allocation. There is no training sequence in this burst for reducing signaling complexity, but a fixed synchronization pattern is used. The access burst is used to determine the distance of the mobile from the base station. This scenario arises in two basic cases. Firstly, this happens when the mobile makes the first attempt for signaling (e.g. Location Update) or for communication (e.g. mobile originated call or SMS). The second scenario occurs when the mobile moves from one cell to another (e.g. handover). In this case the first burst to the new base station is the access burst and it is used to synchronize the mobile with the new base station.

Higher symbol rate burst (HB): this burst is used to carry information on full rate packet data traffic channels using higher symbol rate. It contains 138 encrypted symbols and includes a guard time of 10.5 reduced symbol periods.

2.2.8 Idle mode

The idle mode behavior is managed by the MS. It can be controlled by parameters which the MS receives from the base station on the Broadcast Control Channel (BCCH). All of the main controlling parameters for idle mode behavior are transmitted on the BCCH carrier in each cell. The idle mode tasks can be divided into 4 processes: Public Land Mobile Network (PLMN) selection, cell selection, cell re-selection, and location updating. PLMN selection is the process in which the MS tries to connect or associate with a Public Network after being powered on or after experimenting lack of coverage [11]. In this section we will not explain in detail PLMN selection, since the cases where it takes place are not within our area of interest because either they require active user intervention or they occur in very special conditions, but not normal conditions when a person moves in an indoor scenario.

2.2.8.1 Cell selection

The cell selection algorithm tries to find the most suitable cell of the selected PLMN according to various requirements. If no suitable cell is found and all available and permitted PLMNs have been tried, the MS will try to camp on a cell irrespective of PLMN identity and enter a limited service state. In this state the MS will be able to make emergency calls only. If the MS loses coverage it will return to the PLMN selection state and select another PLMN. There are two different strategies to perform cell selection: normal cell selection or stored cell list selection [11]. In this section we will explain normal cell selection since it is the most used.

The choice of such a suitable cell for the purpose of receiving normal service is referred to as "normal camping". There are various requirements that a cell must satisfy before an MS can perform normal camping on it [11]:

1) It should be a cell of the selected PLMN or, if the selected PLMN is equal to the last registered PLMN, an equivalent PLMN.

2) It is not barred (when a cell is barred it will not be camped on by an MS in idle mode but a MS in dedicated mode can perform handover to it).

3) It does not belong to a location area included in the list of "forbidden location areas for roaming". The location areas that are forbidden, after an attempt to do a location updating has failed, will be stored in the MS as forbidden location areas for national roaming. This list will be cleared when the mobile is powered off.

4) The radio path loss between MS and BTS must be below a threshold set by the PLMN operator.

5) It should not be a SoLSA (Cells on which normal camping is allowed only for MS with Localized Service Area subscription) exclusive cell to which MS does not subscribe. This requirement is only valid for MSs supporting SoLSA.

Initially, the MS looks for a cell which satisfies these 5 constraints ("suitable cell") by checking cells in descending order of received signal

strength. If a suitable cell is found, the MS camps on it and performs any registration necessary. Cells can have two levels of priority; suitable cells which are of low priority are only camped on if there are no other suitable cells of normal priority.
In order to speed up these processes, a list of the RF channels containing BCCH carriers of the same PLMN is broadcast in the system information messages. Also, the MS does not need to search all possible RF channels to find a suitable cell [11].

When the MS has no information on which BCCH frequencies that are used in the network, the MS will search all RF channels in its supported frequency band, take measurement samples of the received RF signal strength and calculate the received average level for each. The average is based on at least five samples per RF carrier spread evenly over a 3 to 5 second period. The MS searches at least the number of the strongest RF channels in descending order of RSS to see which ones are BCCH carriers. If no BCCH carriers have yet been found, searching will continue until at least one BCCH carrier is found. The first BCCH carrier found which is from a suitable cell and on which there is a normal priority indication is taken and that cell is camped on. If at least the numbers of the strongest RF channels have been tried and the only suitable cells found have low priority indication the MS shall camp on the strongest of these cells. If at least the 30 strongest GSM 800 or GSM 900 RF channels or 40 strongest GSM 1800 RF channels or 40 strongest GSM 1900 RF channels have been tried and no suitable cell was found, the MS will select another PLMN according to the PLMN selection procedure and search for suitable cells there [11].

2.2.8.2 Cell re-selection

After a cell has been successfully selected, the MS will start the cell reselection tasks. It will continuously make measurements on its current serving cell and neighboring cells, in order to initiate cell reselection if necessary. The following events trigger a cell reselection [11]:

- The path loss to the cell has become too high.
- There is a downlink signaling failure.
- The cell camped on (current serving cell) has become barred.
- There is a better cell in terms of the path loss criterion in the same registration area, or a much better cell in another registration area of an equivalent PLMN.

- Upper layers have determined that the network has failed an authentication check.

The MS continuously monitors all neighboring BCCH carriers, as indicated by the BA list, in addition to the BCCH carrier of the serving cell, to detect if it is more suitable to camp on another cell. At least five received signal level measurement samples are required for each defined neighboring cell. A running average of the received signal level will be maintained for each carrier in the BA list. All system information messages sent on BCCH must be read at least once every 30 seconds in order to monitor changes in cell parameters. The MS also tries to synchronize to and read the BCCH information that contains parameters affecting cell reselection for the six strongest non-serving carriers (in the BA list) at least every five minutes. The MS also attempts to decode the BSIC parameter for each of the six strongest cells, to confirm that it is still monitoring the same cells [11].

Before camping on the cell after re-selection, the MS shall attempt to decode the full set of system information. The MS shall check that the parameters affecting cell re-selection are unchanged. If a change is detected the MS shall check if the cell re-selection criterion is still valid using the changed parameters. If the cell selection criteria are still valid, the MS shall camp on the cell. If they are not still valid, the MS shall repeat this process for the cell with the next highest RSS [11].

2.2.8.3 Location updating

To make it possible for the mobile subscriber to receive a call, the network must know where the MS is located. To keep the network updated on the location of the MS, the system is informed by the MS on a regular basis. This is called Location Updating. There are three different types of location updating defined: normal, periodic registration and International Mobile Subscriber Identity (IMSI) attach/detach [12].

Normal location updating: the location update is initiated by the MS when it detects that it has entered a new location area. The MS listens to the system information, compares the Location Area Identity (LAI) to the one stored in the MS on the SIM card (on BCCH channel if idle or SACCH channel if active) and detects whether it has entered a new location area or is still in the same location area. If the broadcast LAI differs from the one

stored on the SIM card, the MS must perform a normal location update following the next steps. [12]

1) The MS sends a channel request message including the reason for the access. Reasons other than location updating can be for example, answering a page or emergency call.

2) The message received by the BTS is forwarded to the BSC. The BSC allocates an SDCCH, if there is one idle, and tells the BTS to activate it.

3) The MS is now told to tune to the SDCCH.

4) The MS sends a location updating request message that contains the identity of the MS, the identity of the old location area and the type of updating.

5) The authentication parameter is sent to the MS. In this case the MS is already registered in this MSC/VLR and the authentication parameter used is stored in the VLR. (If the MS is not already registered in this MSC/VLR the appropriate HLR or the previously used MSC/VLR must be contacted to retrieve MS subscriber data and authentication parameters).

6) MS sends an answer calculated using the received authentication parameter.

7) If the authentication is successful, the VLR is updated. If needed, the HLR and old VLR are also updated.

8) The MS receives an acceptance of the location updating.

9) The BTS is told to release the SDCCH.

10) The MS is told to release the SDCCH and switches to idle mode

Periodic Registration location updating: to reduce unnecessary paging of a mobile that has left the coverage area, has run out of battery power or for any other reason has the wrong status in the Mobile Switching Center (MSC)/Visitor Location Register (VLR), there is a type of location

updating called periodic registration. The procedure is described below [12]:

1) MS listens on the BCCH to specify if Periodic Registration Location Update is used in the cell. If periodic registration is used, the MS is told how often it must register. The time is set by the operator and can have values from 0 to 255 deci-hours (a unit of six minutes). If the parameter is equal to zero, periodic registration is not used in this cell. If the parameter is set to ten, for example, the MS must register every hour.

2) Both the MS and the MSC have the timer which controls the procedure. When the timer in the MS expires, the MS performs a location updating, type periodic registration. After that, the timers in MS and MSC restart. In the MSC there is a time scanning function for the MSs. If the MS does not register within the determined interval plus a guard time, then the scanning function in MSC detects this and the MS is flagged as detached.

This process is controlled by the T3212 parameter, which is a timeout value broadcast to the MS in the system information messages. The interval ranges between six minutes (T3212 = 1) and 25.5 hours (T3212 = 255). The periodic registration timer is implemented in the MS, and it will be reinitiated every time the MS returns to idle mode after being in dedicated mode [12]

IMSI attach/detach: The IMSI attach/detach operation is an action taken by an MS to indicate to the network that it has entered into idle mode/inactive state. When an MS is powered on, an IMSI attach message is sent to the MSC/VLR. When an MS is powered off, an IMSI detach message is sent. Therefore, it requires active user intervention and it is not relevant for our approach [12].

2.3 Universal Mobile Telecommunication System (UMTS)

2.3.1 Introduction to WCDMA

WCDMA was chosen as the radio technology for universal mobile telecommunication system (**UMTS**) as is shown in fig. 1. There are two modes of operation frequency division duplex (**FDD**) and time division

duplex (**TDD**). In the FDD, the uplink and downlink uses different frequency bands meanwhile in the TDD, the uplink and downlink share a frequency band by using synchronize intervals. Our work focuses on UTRA-FDD uplink as it is mainly used in Europe and we want to analyze the signal quality in the reverse link [13].

2.3.2 UMTS system architecture

The three main components of UMTS is the user equipment, the UMTS terrestrial radio-access network (**UTRAN**), and the core network. Furthermore, there are two interfaces, the Iu interface between the core network and the UTRA network, and the Uu radio interface between the UTRA network and the user equipment. The UTRAN comprises some of the following functions: encryption/decryption, power control, modulation, multiplexing, etc [13].

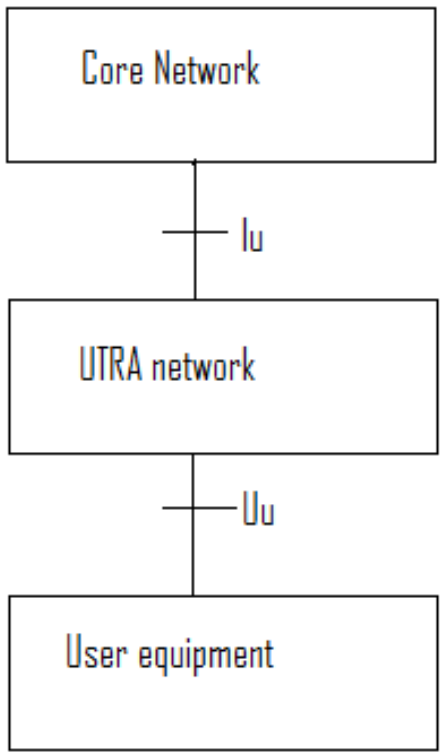


Fig. 1. Components of the UMTS system

2.3.3 UMTS radio interface

WCDMA uses a basic chip rate of 4.096 Mchip/s and different spreading factors. The different rates that UMTS can support are due to the fact of different spreading factors. As it is implied by UTRAN-FDD, uplink and downlink use different frequencies. For Europe, the uplink uses the frequencies between 1920 and 1980 MHz and the base station uses the frequencies between 2110 to 2170 MHz.

2.3.4 Physical structure

There are two types of dedicated channels: the dedicated physical data channel (**DPDCH**), which carries data generated at layer two or above; and the dedicated physical control channel (**DPCCH**), which carries control information at layer 1. Each connection has one DPCCH and one or many DPDCH's.

There are some common defined channels: primary and secondary common control physical channels (CCPCH). They carry downlink information, synchronization channel (**SCH**), which are used for cell search and the physical random access channel (**RACH**) [14].

Fig. 2 shows the structure of a UTRA-FDD (WCDMA) frame. It can be seen on the figure that a radio frame comprises 15 time slots of length 10 ms. Time slots are used to support periodic functions whereas GSM slots are used for user separation. Each time slot contains 2,560 chips. The bandwidth per WCDMA channel is 4.4 to 5 MHz. The DPDCH carries layer 2 data, while the DPCCH carries pilot bits, transmit-power control (**TPC**) commands, the feedback information field (**FBI**) and the transport format combination identifier (**TFCI**) [9].

The DPCCH and DPDCH are mapped to I and Q components and spread to the chip rate with two different channelization codes. The resulting complex signal is scrambled and QPSK modulation with root-raised cosine pulse shaping with rolloff factor of 0.22 is applied in the frequency domain. Orthogonal variable spreading factor (OVSF) codes are used for the channelization which they are used to preserve orthogonality between physical channels with different rates and spreading factors. Channelization codes and UE-specific scrambling codes are given by the network.
Idle mode and uplink channels are similar to GSM.

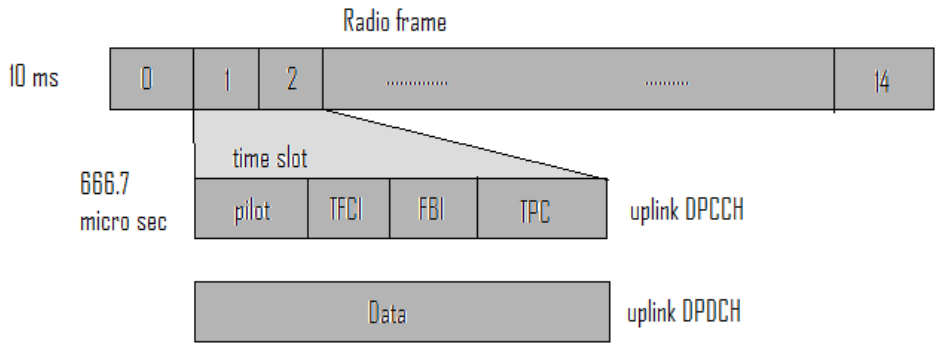


Fig. 2. Uplink frame structure

2.4 IEEE IEEE 802.11

2.4.1 Overview

The IEEE 802.11 family consists of a series of over-the-air modulation techniques that use the same basic protocol. Wireless Local Area Networks (WLAN) based on IEEE 802.11 are been deployed with a great success on a great variety of home, office and corporate environments. Since the introduction of the IEEE 802.11 standard, multiple extensions and amendments have been proposed and accepted, but the most popular and worldwide spread are those defined by the IEEE 802.11b and IEEE 802.11g protocols, and therefore they will be treated in this section.

Both standards work in the band of 2.4 GHz, and they both occupy 13 channels separated 5 MHz with a bandwidth of 22 MHz. The different channels can be seen in table 1.

2.4.2 IEEE IEEE 802.11b Physical layer

IEEE IEEE 802.11 uses two different spread spectrum techniques in its physical layer, which are incompatible:
Direct-Sequence Spread Spectrum (DSSS): DSSS spreads the data across a broad range of frequencies using a high-rate code sequence with binary phase shift keying (BPSK) to modulate the carrier for spreading.

TABLE 1. LIST OF IEEE 802.11 B/G CHANNELS [13]

Channel	Center Frequency (GHz)	Channel Width (GHz)	Overlapped Channels
1	2.412	2.401 - 2.423	2,3,4,5
2	2.417	2.406 - 2.428	1,3,4,5,6
3	2.422	2.411 - 2.433	1,2,4,5,6,7
4	2.427	2.416 - 2.438	1,2,3,5,6,7,8
5	2.432	2.421 - 2.443	1,2,3,4,6,7,8,9
6	2.437	2.426 - 2.448	2,3,4,5,7,8,9,10
7	2.442	2.431 - 2.453	3,4,5,6,8,9,10,11
8	2.447	2.436 - 2.458	4,5,6,7,9,10,11,12
9	2.452	2.441 - 2.463	5,6,7,8,10,11,12,13
10	2.457	2.446 - 2.468	6,7,8,9,11,12,13
11	2.462	2.451 - 2.473	7,8,9,10,12,13
12	2.467	2.456 - 2.478	8,9,10,11,13
13	2.472	2.461 – 2.483	9,10,11,12

This is combined with lower rate data that also modulates the carrier with either differential BPSK (DBPSK) or differential quadrature PSK (DQPSK). The high-rate code is an 11-bit Barker code that provides 11 MHz chipping rate and has good autocorrelation properties and gives waveform protection to interference and multipath. The data rate varies between 1, 2, 5.5 and 11 Mb/s (DBPSK and DQPSK modulation). The typical receiver can operate with a 0 dB signal-to-noise ratio in the spread bandwidth and therefore can tolerate strong multipath and delay spread. DSSS uses a lower power density (power/frequency), making it harder to detect. DSSS also sends redundant copies of the encoded data to ensure reception. Narrowband interference appears to the receiver as another narrowband transmission. When the total received signal is decoded, the wider band transmission (DSSS encoded data) is decoded with the same code back to its original narrowband format while the interference is decoded to a lower power density signal, thereby reducing its effects. When broadband interference is present, however, the resulting decoded broadband interference can give a much higher noise floor, almost as high as the decoded signal. For this reason, DSSS works better for large data packets in a low to medium interference environment, but not as well in higher interference industrial applications [15].

Frequency Hopping Spread Spectrum (FHSS): in this method the 2.4 GHz band is divided into 79 channels of 1MHz, each one modulated with Gaussian Frequency Shift Modulation (GFSK). Sender and receiver match the hop pattern of switching the channel and data is transmitted in different channels according to the frequency pattern. Each data transfer in implemented according to a different hop pattern, and the hop patterns are developed so as to minimize the probability of usage of the same channel simultaneously by two users. FHSS is a robust technology, with good behavior in harsh environment characterized by large areas of coverage, multiple collocated cells, noises, multipath or Bluetooth presence, but it is limited to just 2 Mb/s data rate, which makes it less popular than DSSS [16].

2.4.3 IEEE IEEE 802.11g Physical layer

IEEE IEEE 802.11g uses Orthogonal Frequency Division Multiplexing (OFDM), a Frequency-Division Multiplexing (FDM) scheme used as a digital multi-carrier modulation method. A large number of closely-spaced and overlapped orthogonal sub-carriers are used to

carry data. By carefully specifying the symbol duration and frequency spacing, subcarriers will generate no interference or distortion on adjacent subcarriers. The data is Viterbi encoded prior to transmission and distributed across several parallel data streams carrying by 52 sub-carriers via an interleaver. Each sub-carrier is modulated with a conventional modulation scheme (such as QAM or PSK) at a low symbol rate, maintaining total data rates similar to conventional single-carrier modulation schemes in the same bandwidth. After modulation, N parallel symbol streams (at 1/N of the original rate) are fed to the N point Inverse Fast Fourier Transformer (IFFT). After IFFT, a cyclic prefix (CP) is added at the beginning of the symbol to make the signal robust against multipath propagation [17].

The IEEE 802.11g OFDM symbol rate is 250 KHz, corresponding to a symbol period of 4 microns. The OFDM pulse contains a guard interval, which is 800ns in duration. During the guard interval, inter-symbol interference (ISI) from a delayed pulse arriving via a delayed path may occur. However, once the signal arrives at the receiver and is translated to the digital domain, the guard interval is essentially discarded by the baseband processor. With the guard interval eliminated, a 3.2s rectangular FFT window remains. The zero-ISI condition in the frequency domain is met due to the use of a 3.2s FFT window in conjunction with 312.5 KHz subcarrier spacing. In this manner, cross-talk among subcarriers is eliminated [17].

IEEE IEEE 802.11 g OFDM offers a variety of data rates which covers 6, 9, 12, 18, 24, 36, 48, and 54 Mbit/s, depending on the modulation and coding rate as shown on table 2.

TABLE 2. DATA RATES IN IEEE 802.11G

Data Rate (Mb/s)	Modulation	Coding rate	Coded bits per sub-carrier	Coded bits per OFDM symbol	Data bits per OFDM symbol
6	BPSK	½	1	48	24
9	BPSK	¾	1	48	36
12	QPSK	½	2	96	48
18	QPSK	¾	2	96	72
24	16-QAM	½	4	192	96
36	16-QAM	¾	4	192	144
48	64-QAM	2/3	8	288	192
54	64-QAM	¾	8	288	216

2.4.4 DSSS-OFDM

Because of the quick success of IEEE 802.11g, the IEEE implemented a mode where IEEE 802.11b and IEEE 802.11g could coexist in the same network. With this functionality, an IEEE 802.11g network can admit IEEE 802.11b terminals and vice versa. This hybrid modulation has the DSSS preamble and header and the OFDM payload. Although both standards can coexist in the same network and their hardware is fully compatible, the overall performance decreases significantly in a mixed network. Whereas IEEE IEEE 802.11b reaches up to 11 Mb/s, IEEE IEEE 802.11g can reach up to 54 Mb/s, but its speed decreases considerably when there exists legacy IEEE 802.11b participants in the network [18].

2.4.5 Operating modes

Infrastructure mode*:* in this mode a set of wireless terminals are connected to an access point (AP) forming the wireless network and the AP is connected to a wired network. Such combinations are called Basic Service Set (BSS), and two or more BSS form an Extended Service Set (ESS). Since most of the wireless stations need to access a server, a printer or to Internet available in a wired LAN, this mode is the most used [16].

Ad-Hoc: in this mode the wireless stations are connected directly to each other forming a simple network, without the intervention of an AP [16].

2.4.6 IEEE IEEE 802.11 MAC protocol

The IEEE IEEE 802.11 defines two basics methods to access the medium:

Distributed Coordination Function (DCF): The fundamental access method of the IEEE IEEE 802.11 is a DCF known as a Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA). For a station (STA) to transmit, it shall sense the medium to determine if another STA is transmitting. This is doing by detecting a minimum duration called DCF Interframe Space (DIFS) to see if there is any other transmission in progress on the wireless medium. Afterwards, according to the current Contention Window (CW), the station keeps further sensing the medium by a randomized multiple of a slot time from 0 to (CW-1) to minimize the chance of collision. During the random backoff interval, if the station senses the medium becomes busy, it stops decrement the time counter and does not reactivate the paused value until the channel is sensed idle again for more than a DIFS. The size of CW is controlled by an exponential back-off mechanism. The value is set equal to a pre-specified minimum contention window, CWmin, at the first transmission attempt. The CW is doubled up to maximum contention window, CWmax, if the transmission fails. Once the backoff timer expires, the transmitting station has two options to initiate the transmission. It can either transmit the data packet directly or transmit a short RTS (Request-To-Send) frame, followed by a CTS (Clear-To-Send) frame from the receiving station. Furthermore, the RTS-CTS frame contains the information of how long it takes to transmit the data packet, which can help other stations to understand that they should not attempt to transmit during such period. Once the data packet is delivered to the destination station, an acknowledge (ACK) frame will be sent back to confirm the transmission [18][19].

Point Coordination Function (PCF): The IEEE IEEE 802.11 MAC also incorporates an optional access method called a PCF, which is only usable on infrastructure network configurations. If the PCF is to be used, time is divided into superframes where each superframe consists of a contention period (CP) in which DCF is used and a contention-free period (CFP) where PCF is used. This access method uses a point coordinator (PC), which operates at the access point of the BSS, to determine which STA

currently has the right to transmit. The operation is done with the PC performing the role of the polling master. The CFP is started when the PC distributes information (using the ordinary DCF) within Beacon management frames to gain control of the medium by setting the network allocation vector (NAV) in STAs. During the CFP, the PC polls each station in its polling list (the high priority stations), when they are clear to access the medium. Stations that are polled must always respond to a poll. A station being polled is allowed to transmit a data frame, and in case of unsuccessful transmission the station may retransmit the frame after being re-polled or during the next CP. If there are no pending transmissions, the response from the STA is a null frame containing no payload. To ensure that no DCF stations are able to interrupt this mode of operation, the interframe space between PCF data frames (PIFS) is shorter than the DIFS. To prevent starvation of stations that are not allowed to send during the CFP, there must always be room for at least one maximum length frame to be sent during the CP [19].

2.4.7 Management frames

The purpose of management frames is to establish initial communications between stations and access points. Below we describe the management frames relevant for our research:

Beacon frame: in an infrastructure network, an access point periodically sends a beacon (according to the BeaconPeriod parameter) that provides synchronization among stations utilizing the same BSS The beacon includes a timestamp that all stations use to update what IEEE 802.11 defines as a timing synchronization function (TSF) timer. If the access point supports the point coordination function, then it uses a beacon frame to announce the beginning of a contention-free period. If the network is an independent BSS, all stations periodically send beacons for synchronization purposes. The beacons include information such as SSID, supported rates and security parameters [19]. The beacon packets can be seen on Fig 3 and fig 4 as are capture by Wireshark.

Fig 3 and fig 4 show the beacons transmitted by different Access Points identified by their MAC addresses, with different sequence numbers and flags.

From these flags, information such as supported rates, current channel,

Traffic Indication Map, country or vendor of the AP can be obtained. Additional parameters such as operating mode of the AP, type of network, CFP coordination capabilities, privacy or modulation used are also depicted.

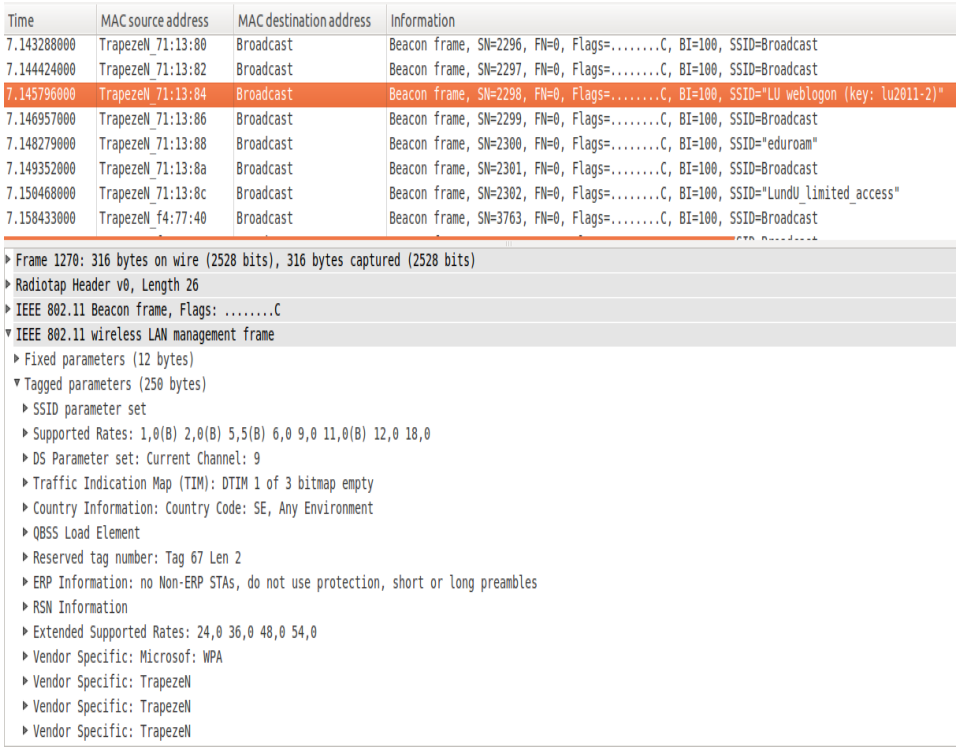


Fig. 3. Beacon tagged parameters

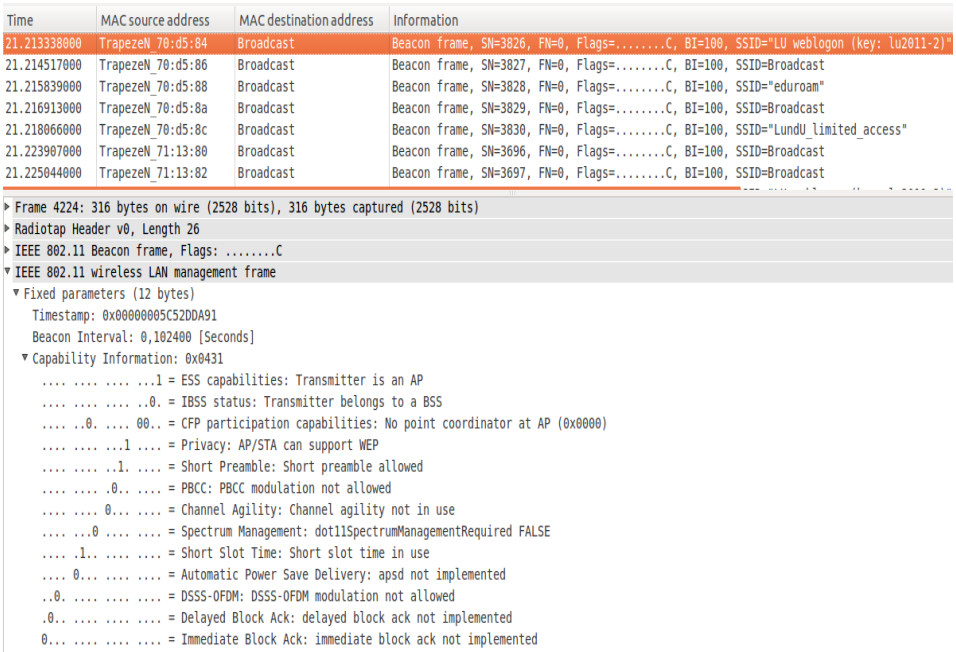


Fig. 4. Beacon fixed parameters

ATIM frame: a station with frames buffered for other stations sends an announcement traffic indication message (ATIM) frame to each of these stations during the ATIM window, which immediately follows a beacon transmission. The station then transmits these frames to the applicable recipients. The transmission of the ATIM frame alerts stations in sleep state to stay awake long enough to receive their respective frames [19].

Authentication frame: a station sends an authentication frame to a station or access point that it wants to authenticate with. The authentication sequence consists of the transmission of one or more authentication frames, depending on the type of authentication being implemented (open system or shared key) [19].

De-authentication frame: a station sends a de-authentication frame to a station or access point with which it wants to terminate secure communications [19].

Probe request frame: a station sends a probe request frame to obtain information from another station or access point. For example, a station may send a probe request frame to determine whether a certain access point is available [19].

Probe response frame: if a station or access point receives a probe request frame, the station will respond to the sending station with a probe response frame containing specific parameters about itself (such as parameter sets for the frequency hopping and direct sequence PHYs)[19].

Association request frame: a station will send this frame to an access point if it wants to associate with that access point. A station becomes associated with an access point after the access point grants permission [19].

Association response frame: after an access point receives an association request frame, the access point will send an association response frame to indicate whether or not it is accepting the association with the sending station [19].

Re-association request frame: a station will send this frame to an access point if it wants to re-associate with that access point. A re-association may occur if a station moves out of range from one access point and within range of another access point. The station will need to re-associate (not merely associate) with the new access point so that the new access point knows that it will need to negotiate the forwarding of data frames from the old access point [19].

Re-association response frame: after an access point receives a re-association request frame, the access point will send a re-association response frame to indicate whether or not it is accepting the re-association with the sending station [19].

2.4.1.6 Management procedures

Active scanning: consists on issuing **probe request** frames to which APs will respond with probe **response frames** including information similar to the information included in beacon frames. This way, the STA collects information about candidate APs and chooses one of them. The selection of which access point to use depends on several parameters such as quality of signal, access network capabilities, user preferences and policy [19].

Passive scanning: the STA listens for **beacons** frames issued by the APs at regular intervals [19].

Association: after a successful scanning in infrastructure mode, the STA will try to associate with an AP. The STA will try to authenticate itself by sending an authentication request to the AP, and the AP will respond with an authentication response. If the authentication process is successful, the STA sends an association request to the AP, and the AP responds with an association response. If the STA meets the requirements to join the BSS that the AP belongs to, the AP will include in the association response a status "successful". After the STA replies with an ACK, it will be associated with the specific AP and it will be able to send traffic [19].

Re-association: when a station moves out of coverage of its associated AP, it performs handoff procedures via scanning new APs and re-associating with another AP. The STA initiates the same process as the one described in the association procedure, but in this case the STA sends a re-association request frame including the MAC address of the current AP, so that the new AP can use it to communicate with the "old" AP for handoff procedures [19].

Fig 5 shows the packets exchanged by an Access Point and a mobile device during active scanning and association procedures.

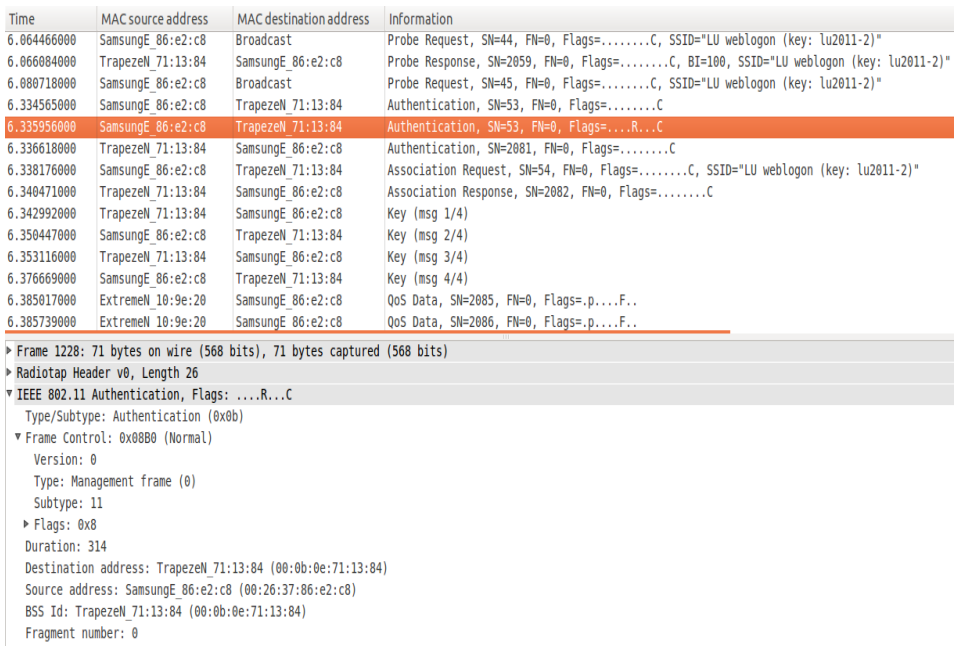


Fig. 5. Scanning and association procedures

The active scanning is performed by a Samsung mobile device, which sends out a Probe Request. The Access Point with the MAC address Trapezen_71:13:84 responds (the first 3 bytes of the MAC address correspond to the name of the device vendor) affirmatively with a Probe Response, and the authentication process is started. After successful authentication the mobile device requests association with the AP by sending an Association Request, to which the AP responds positively with an Association Response. Then, the AP asks the mobile device for the key of the network, and once the mobile device provides this key (in this case is automatically sent as it is stored in the mobile device memory), the data traffic exchange starts.

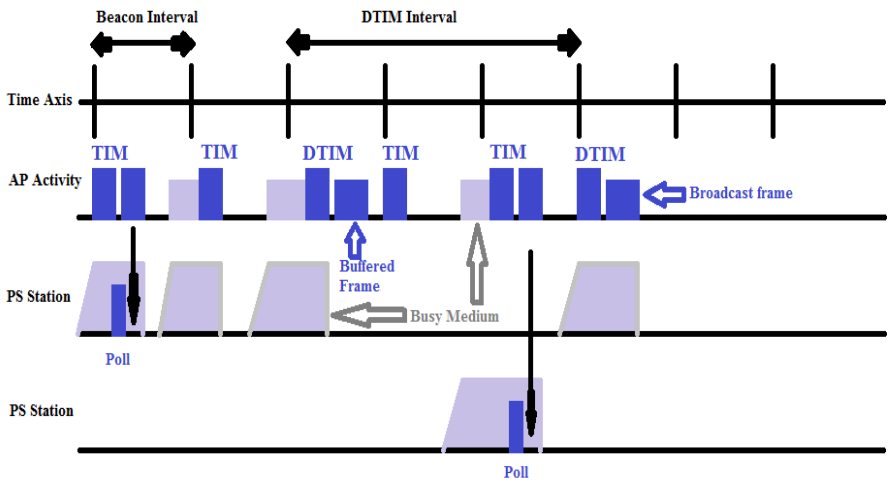
2.4.2 Power saving mode (PSM)

PSM is the functionality that IEEE 802.11 networks utilize when there is no active data traffic between the members of the network. The objective of the IEEE 802.11 PSM is to let the wireless interface of a mobile host in the active mode only for the time necessary to exchange data, and to turn it in sleep mode whenever it becomes idle. As this master thesis deals with idle modes in wireless networks, this is an important issue in our research.

Each mobile host within the hotspot informs the Access Point on whether it utilizes the PSM or not by using the Power Management bits within the Frame Control field of transmitted frame. Since the Access Point relays every frame from/to any mobile host, it buffers the frames addressed to mobile hosts using the Power-Saving Mode, and only transmit them at designated times. PSM mobile hosts are synchronized with the Access Point, and wake up to receive Beacons. The STAs that currently have buffered MAC Service Data Units (MSDUs) within the AP are identified in a traffic indication map (TIM), which is included as an element within all beacons generated by the AP. The TIM indicates PSM mobile hosts having at least one frame buffered at the Access Point, and a STA determines that an MSDU is buffered for it by receiving and interpreting this TIM**.** This information is coded in a partial virtual bitmap, whose bits are set by the AP to select the STAs to which it has to forward pending buffered MSDUs. If any STA in its BSS is in PS mode, the AP buffers all broadcast and multicast MSDUs and deliver them to all STAs immediately following the next Beacon frame containing a delivery TIM (DTIM) transmission. This

process is done using normal frame transmission rules, and before transmitting any unicast frames to specific STAs [18].

If a STA is indicated in the TIM, it “downloads” the frames by sending a special frame (PS Poll) to the Access Point by means of the standard DCF procedure. Upon receiving a PS-Poll, the Access Point sends the first DATA frame to the PSM mobile host, and receives the corresponding ACK frame. If appropriate, the Access Point sets the More Data bit in the DATA frame, to announce other frames to the same PSM mobile host. To download the next frame, the mobile host sends another PS-Poll. If the TIM indicating the buffered MSDU is sent during a contention-free period (CFP), a CF-Pollable STA operating in the PS mode does not send a PS-Poll frame, but remains active until the buffered MSDU is received (or the CFP ends). A STA shall remain in its current PSM until it informs the AP of a PSM change via a successful frame exchange. When, eventually, the mobile host has downloaded all the buffered frames, it switches to the sleep mode [18]. Fig 6 shows a typical PSM scenario.



.

Fig. 6. PSM scenario with DTIM at every 3 TIM [9]

CHAPTER 3

3 GSM/UMTS analysis

3.1 Brief description of USRP

The Universal Software Radio Peripheral (USRP) allows you to create software radios using a computer with a USB 2.0, or Gigabit Ethernet port regarding the version of the USRP. There are many daughter boards on different radio frequency bands that can be used. For GSM and UMTS, we used the DBSRX, which is a daughter board that can receive either GSM or UMTS (800 – 2400 MHz). The prices of USRP's vary regarding the characteristics from $700 to $1700 dollars and daughter boards from $75 to $475 dollars. The prices were taken based on [22]

The USRP works with GNU Radio where you can create your own applications written in python programming language and the signal processing is in C++.

For more detailed description of USRP's characteristics and functions refer to appendix 6.

One important limitation is that the USRP is capable of processing signals up to 8 MHz wide. The uplink band for GSM is 25 MHz wide and for UMTS is 60 MHz wide. Hence, if we wanted to collect data in a wider band, we would have to pinpoint certain frequencies in order to listen to either GSM or UMTS signals in a wider band.

The measurements for GSM and UMTS were performed at LTH in the E building. It was observed that the activity located for the mobile phone was within the first12 MHz of the uplink band for GSM (from 890 MHz to 902 MHz). While for UMTS, it was observed that the activity were located at the end of the uplink band, so we scanned the last three channels (1967.4 MHz, 1972.4 MHz and 1977.4 MHz). These pre-measurements mentioned before were performed with the spectrum analyzer.

It was scanned 12 MHz from 890 MHz to 902 MHz. It is important to remember that the USRP can only scan 8 MHz at a time. Hence, it is necessary to tune the center frequency as follows:

- First step is 894 MHz (it will cover frequency band from 890 MHz to 898 MHz).
- Second step is 902 MHz (it will cover frequency band from 898 MHz to 906 MHz).

It was important to use FTT overlapping to reduce the non linearity response of the digital down converter (frequency is not flat from –Fs/2 to +Fs/2). So, it was chosen to use an overlap of 50%, this means that the step size would be 4MHz. Thus, it was necessary to tune the center frequency as follows:

- First center frequency 892.7 MHz (frequency band from 888.7 MHz to 896.7 MHz).
- Second center frequency 896.7 MHz (frequency band from 892.7 MHz to 900.7 MHz).
- Third center frequency 900.7 MHz (frequency band from 896.7 MHz to 904.7 MHz).

To do this, it was used the usrp_spectrum_sense.py (generally found in gnuradio/gnuradio-examples/python/usrp/usrp_spectrum_sense.py). This program is from the GNU radio software. In order to run it a proper command would be ./usrp_spectrum_sense.py –f initial frequency final frequency. The output of the command will be N samples point of the FFT at different center frequencies without FFT overlapping. For more information about the output of the command refer to appendix 6.

For the specific purposes, some lines were added to the script usrp_spectrum_sense.py to support more than one USRP connected to one computer. On the other hand, it was added some functions for FFT overlapping, rearrange the output, and print it in the following order from x[512] to x[1023] followed by x[1] to x[511] :

def adjust_data(m_data,fft_size): It takes the output of the FFT (m_data) and the size (fft_size) as arguments of the function. It returns the output of the FFT from –Fs/2 to +Fs/2.
def overlapping(f_cent_1,f_resol,n_freq,f_step,f_size,t_data): It takes

the first center frequency (f_cent_1), frequency resolution (f_resol) that for the measurements was 7812.5 MHz, the number of center frequencies (n_freq) that will be used to scan the desire spectrum, the frequency step (f_step) that was 4 MHz, the FFT size (f_size) which 1024 was chosen, and the data arranged by adjust_data() (t_data) as arguments of the function. It performs the FFT overlapping (50%) and returns the result.

def write_to_file(data,y): It writes the data in a text file so it can be processed by Matlab.

To collect in a specific radio frequency band, type the following command in the terminal:
./usrp_spectrum_sense.py -g xx --tune-delay xx --dwell-delay xx -F xx -d xx xx.

-g xx: it is the gain (FPGA gain + RF amplifier gain).
-- tune-delay xx: When the USRP changes its center frequency, the USRP has to wait until right number of samples arrive to our FFT engine and be sure that it belongs to the desire center frequency.
--dwell-delay: it is the time that takes to collect N=1024 number of samples with decimation rate = 8 is 128 us.
--F xx: it is the FFT size.
--d xx xx: it is the initial and final frequency.

Finally, in order to collect data for GSM, the following command was executed in the terminal: ./usrp_spectrum_sense.py -g 40 --tune-delay 0.050 --dwell-delay 0.000130 -F 1024 -d 8 888.7M 904M.

In a similar way, it was executed the following command in the terminal for UMTS: ./usrp_spectrum_sense.py -g 40 --tune-delay 0.050 --dwell-delay 0.000130 -F 1024 -d 8 1.9654G 1.9794G.

3.2 Active mode

It is important to check how often the mobile phone sends signals in the uplink channel. In order to reduce user intervention, it was important to study the idle mode state of the mobile phone, which is when the user doesn't interact with the mobile phone.
It was important to verify that the USRP was collecting correct data with all

the modifications done to usrp_spectrum_sense.py. The easiest way to prove it was during the active mode state. It was performed an attempt call for GSM and UMTS from one mobile phone to another and raw data was collected with the USRP.

3.3 GSM and UMTS Attempt call

As it was mentioned before the measurement procedure for an attempt call for either GSM or UMTS was performed as it can be seen in the following picture.

Fig. 7. USRP, mobile phone and host

It is important to remind a couple of technical details for GSM and UMTS. The uplink band goes from 890 to 915 MHZ for GSM. GSM uses slow frequency hopping and in order to overcome this problem for collecting data was measured a wider band range. For the measurements was taken only the first 12 MHz as it was observed activity with the spectrum analyzer within the first 12 MHz for that specific position in that specific geographic area. GSM uses FDMA to operate in various frequencies separated 200 KHz. The GSM spectrum within the first 12 MHz of the uplink band can be seen in fig. 8 for an attempt call. There are many

frequencies in which the signal hop and each carrier frequency is separated 200 KHz each. This can be seen more clearly in fig. 9.

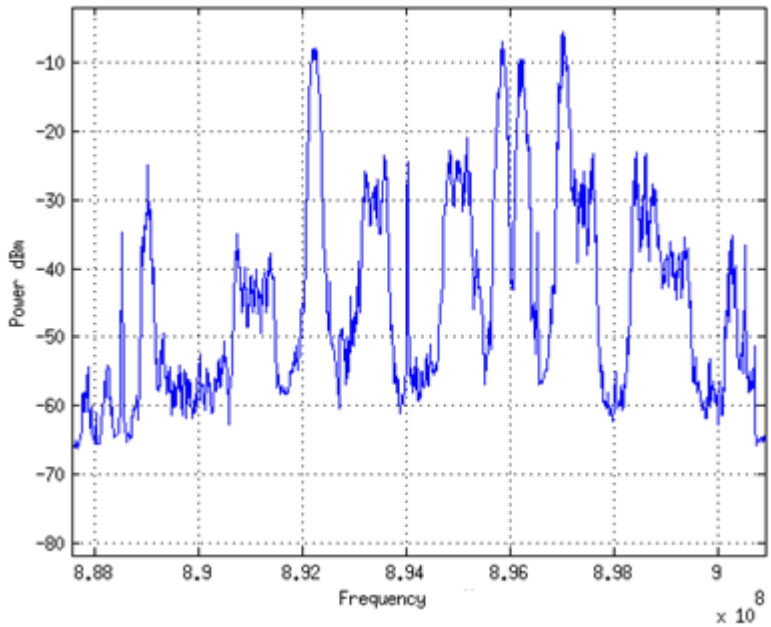


Fig. 8. GSM uplink spectrum

For UMTS, the uplink band is from 1920 to 1980 MHz. Regarding the specific position and geographical area, it was observed that the activity for UMTS was located in the last three channels of the band in the spectrum analyzer (1967.4 MHz, 1972.4 MHz and 1977.4 MHz). Each channel is 5 MHz wide and as it uses spread spectrum, signal energy is distributed along the frequency band.

In the fig. 10 can be seen that the signal activity was located on the carrier frequency1972.4 MHz. In addition, the signal energy is distributed along 5 MHz (+2.5 MHz and -2.5MHz from the carrier frequency). Besides the UMTS spectrum, it can be seen how the signal looks in the time domain. Fig. 11 shows 9.996 seconds of an attempt call. It can be observed clearly where it was signal activity as it produces a change in power levels.

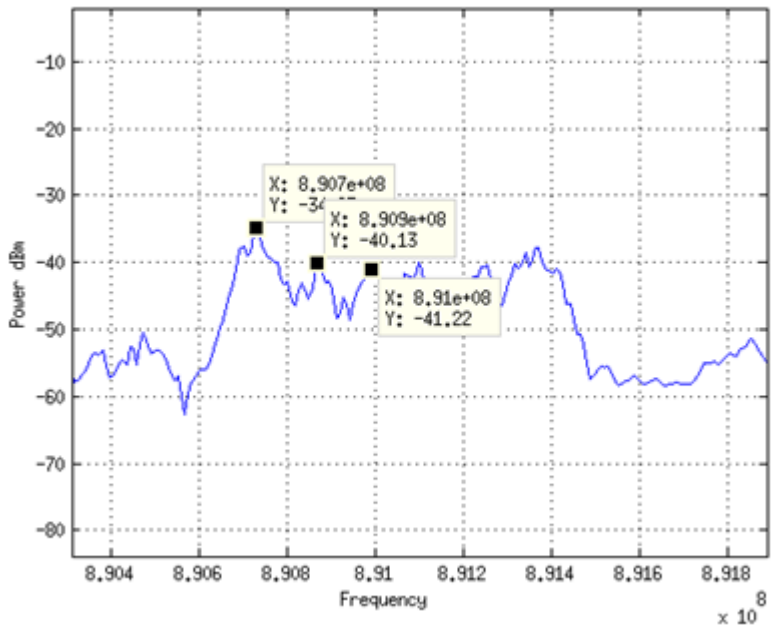


Fig. 9. Frequency carrier separation of 200 KHz for GSM

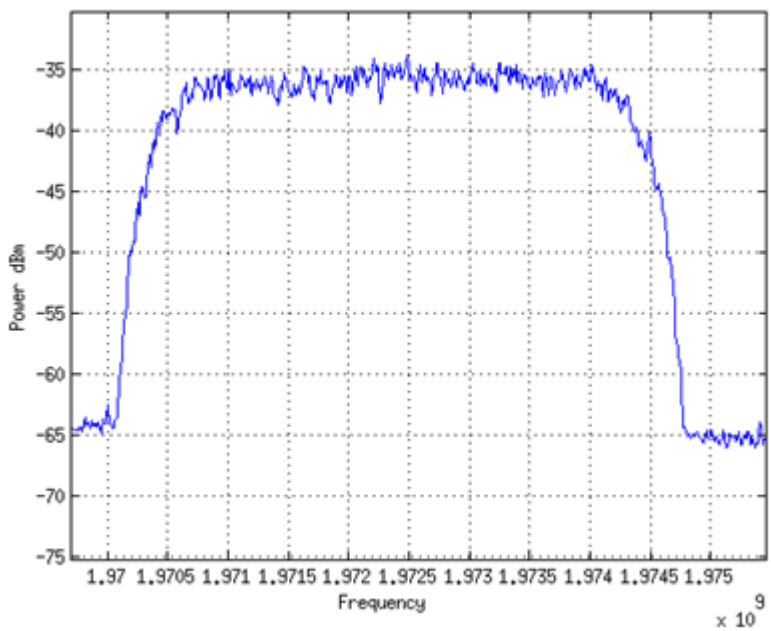


Fig. 10. UMTS frequency spectrum (attempt call)

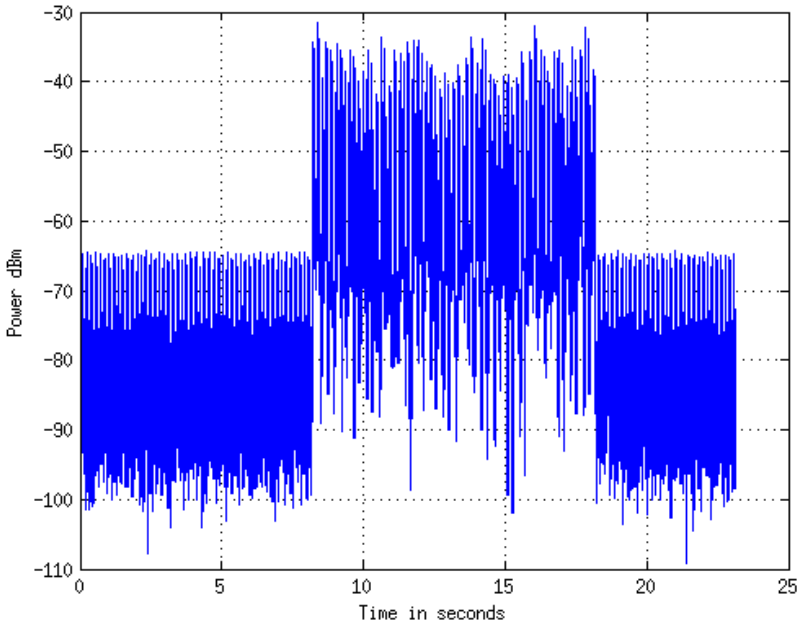


Fig. 11. UMTS time domain (attempt call)

3.4 IDLE mode

In order to reduce user intervention, it was required signal activity during the idle mode. It is important to remember that the idle mode is when the user is not using the mobile phone and there is no activity at all.

There are many tasks involved in the idle mode, furthermore, periodic updates are important to keep the network updated; moreover, as battery life is an important issue in most mobile devices, there has been a big effort to enhanced battery life.

How often periodic signals are triggered is controlled by the network operator. One main disadvantage for the purposes is the lack of regular activity during the idle mode. The time of how often periodic signals are triggered vary from 0 to 255 deci-hours (from six minutes to 25.5 hours).

Fig. 12 shows the signal activity in the idle mode state for GSM during 1858.321283 seconds (approximately 30 minutes), while fig. 13 shows the same, but for UMTS during 1884.056545 seconds (approximately 31

minutes). Based on the fact that there is not continuously signal activity during the idle mode state, we rather focused on WI-FI.

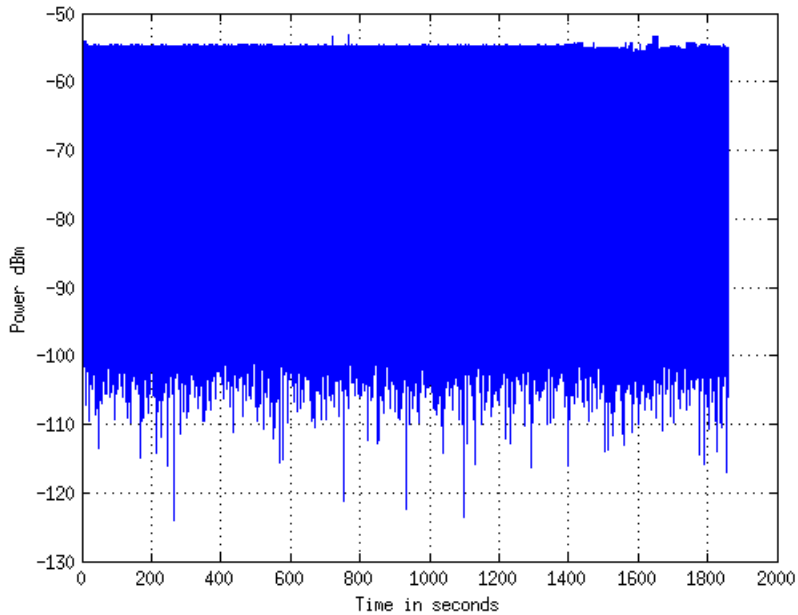


Fig. 12. GSM idle mode

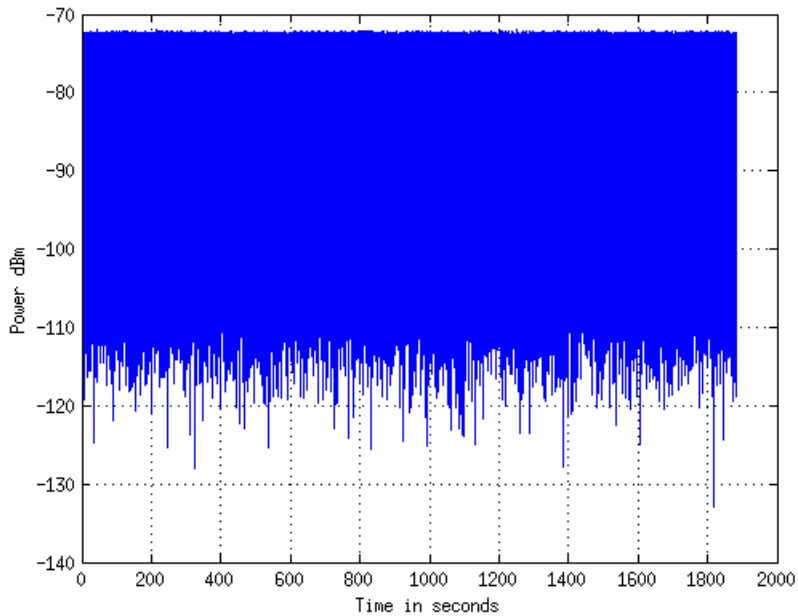


Fig. 13. UMTS idle mode

CHAPTER 4

4 Implementation

4.1 USRP approach

Due to the nature of Wi-Fi, each channel is shared between upstream and downstream. This turns into a challenge of how to separate between the streams as Wi-Fi does not use Frequency Division Multiple Access (FDMA).

Wireshark is a network analyzer that is based on layers 2 and 3. It means that it captures packets and decodes them showing their MAC addresses and power levels. More details about network analyzers will be discussed in 4.2.

The USRP captures incoming signals and collects their power levels. It captures signals on layer 1 (pure physical layer).

In this section the objective is to distinguish between upstream and downstream, and our approach will be to separate them through power levels. It means that if it is placed the USRP close to a router, the USRP will get higher power levels on the downstream than on the upstream.

4.1.1 Detecting IEEE 802.11 signals using USRPs

It is important to remember that Wi-Fi channels for IEEE 802.11 b and IEEE 802.11 g/n are 22 MHz and 20 MHz respectively. Do not forget that USRP can process signals only 8 MHz wide. There is a WI-FI network installed in all corridors along the building. Our measurements were performed at the basement in the E-building; furthermore, the router closest to us was configured to transmit over channel 9.

In order to collect data with the USRP's, it was followed the same method as it was used either for GSM or UMTS. It was just tuned the receiver to

different center frequencies to cover the whole desire bandwidth. For the same purpose, it was used the same code usrp_spectrum_sense.py with proper parameters.

The command to be executed was the following:
./usrp_spectrum_sense.py -g 30 --tune-delay 0.010 --dwell-delay 0.000128 -F 1024 -d 8 2.439G 2.454G

If you wanted to collect data in a different channel you would just need to change the last to parameters of the command to the correct initial frequency and final frequency. For more details check appendix 6.

4.1.2 Scenario of measurements

The assumption was to place the USRP's close to the router in order to get higher power levels coming from the router than from the terminal.

Fig. 14 shows our measurement set up. The USRP1 was located right next to the Wi-Fi access point, the USRP2 3.5 meters away from the USRP1. The distance was limited by the length of the USB. Our hypothesis to get higher power levels coming from the router than from the terminal only applied to USRP1. Due that the USRP2 was 3.5 meters away, power levels coming from the WI-FI access point were similar to the ones coming from the terminal (making the distinction between streams challenging).

4.1.3 Procedure of measurements

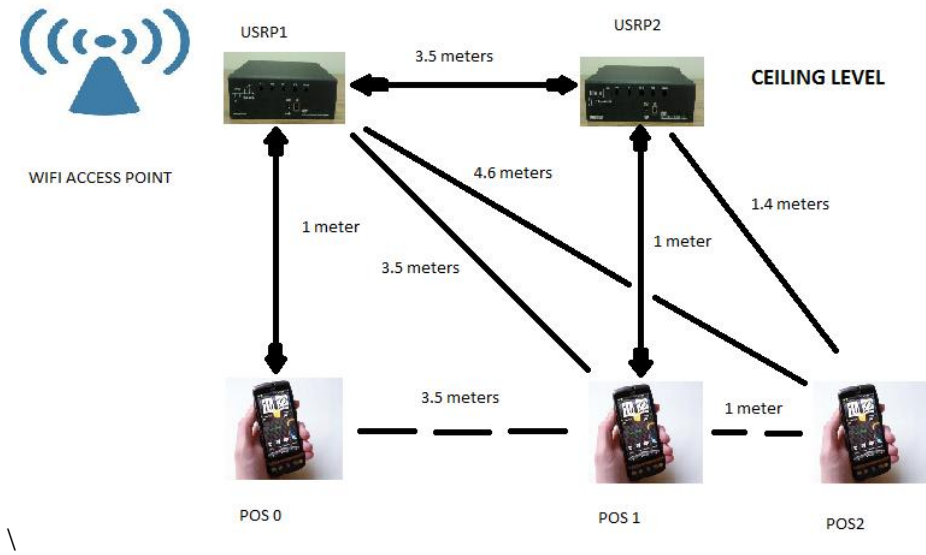
In order to know which power levels were collected from the Wi-Fi access point to both USRP's, all incoming signals were measured to both USRP's to get a reference level of what we called noise (without triggering upstream traffic from our terminals). Two reference levels for each USRP's were obtained.

For USRP1 two thresholds were defined: higher than -40 dBm and lower than -60 dBm was captured most of the noise (downstream traffic), so the power levels between – 40 dBm and -60 dBm was considered for upstream traffic.
For USRP2 two thresholds were defined: higher than -45 dBm and lower

than -55 dBm was captured most of the noise (downstream traffic), so the power levels between -45 dBm and -60 dBm was considered for upstream traffic.

The measurement procedure was the following:

- Stand up on different positions (POS 0, POS 1 and POS 2).
- In each position, attempt a call and record data with both USRP's.
- Regarding the thresholds obtained in the pre-measurements, filter only upstream traffic and plot it.
- Plot in color red upstream traffic and in color blue downstream traffic.



\

Fig. 14. A USRP Measurement set up

Fig. 15 shows the upstream traffic when the mobile phone was in POS 0 according to fig. 14 under USRP1. According to our thresholds for USRP1, the power levels between -40 dBm and – 60 dBm was captured most of the upstream traffic. As the USRP1 was closer to the WI-FI access point than the mobile phone, the USRP1 captured higher power levels on downstream traffic than upstream traffic from the mobile phone.

Fig. 16 shows less upstream traffic captured by USRP1. According on fig. 14, the mobile phone on POS 1 was further compared to POS 0 to USRP1. That means that there might be upstream traffic which was weaker than POS 0 and was not totally captured within our thresholds (between -40

dBm and -60 dBm).

Fig. 17 shows even less upstream traffic captured by USRP1. According on fig. 14, the mobile phone was even further than POS1 to USRP1. Even weaker signals could be captured on POS2 by the USRP1. Many incoming signals detected by the USRP1 were not in our fixed thresholds.

The USRP2 was located according to fig. 14 approximately 3.5 meters away from the Wi-Fi access point. In this case, the USRP2 was not anymore receiving higher power levels from the Wi-Fi access point compare to the upstream traffic. Regarding the position of the mobile phone, upstream traffic could be less, similar or higher than the downstream traffic. This is not good as we have a fix threshold (between -45 dBm and -55 dBm) where upstream traffic should have been.

Fig. 18 and fig. 19 shows the upstream captured by the USRP2 when the mobile phone is on POS 0 and POS 2 according on fig. 14. Notice that when the mobile phone was on POS 1, the USRP2 captured less upstream traffic than POS 0. The reason was that the USRP2 as was not next to the Wi-Fi access point, power levels captured by the USRP2 were not high enough compared to power levels of upstream traffic on POS 1. This can be seen on fig. 20 that there might be upstream traffic that was not within our thresholds and it was considered as downstream traffic.

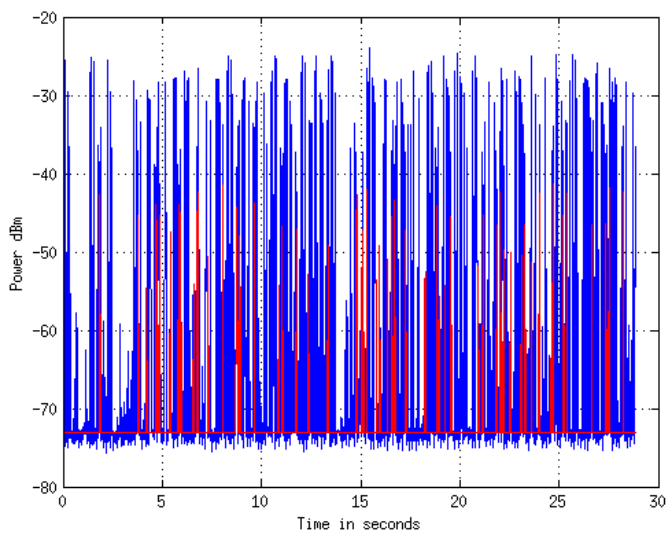


Fig. 15. USRP 1 pos 0, Wi-Fi traffic (Upstream in color red and downstream in color blue)

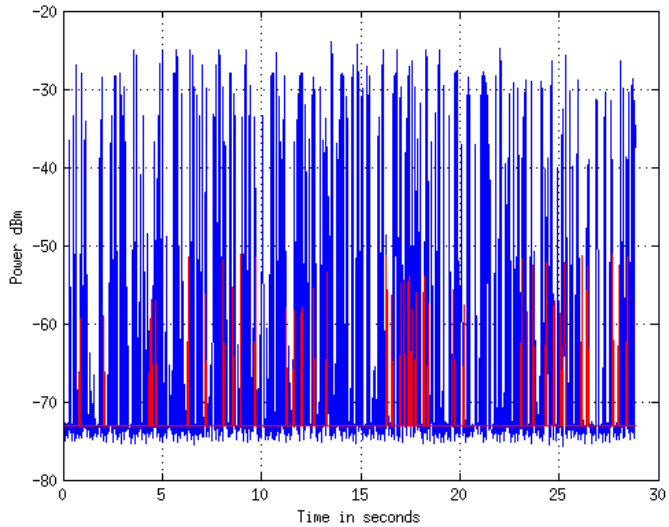


Fig. 16. USRP 1 pos 35, W-Fi traffic (Upstream in color red and downstream in color blue)

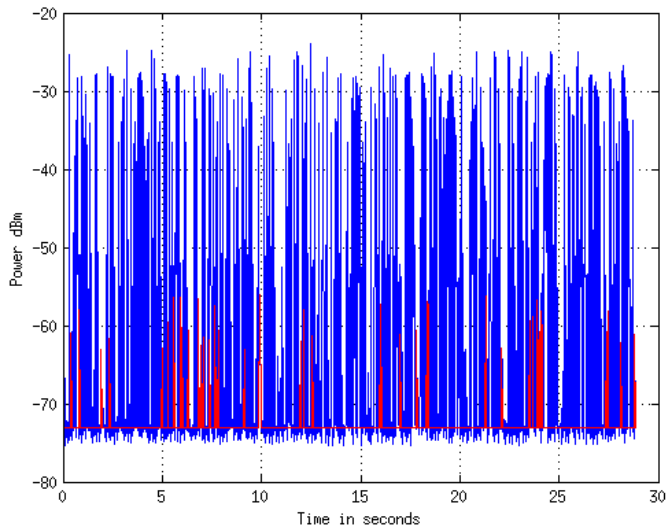


Fig. 17. USRP 1 pos 45, Wi-Fi traffic (Upstream in color red and downstream in color blue)

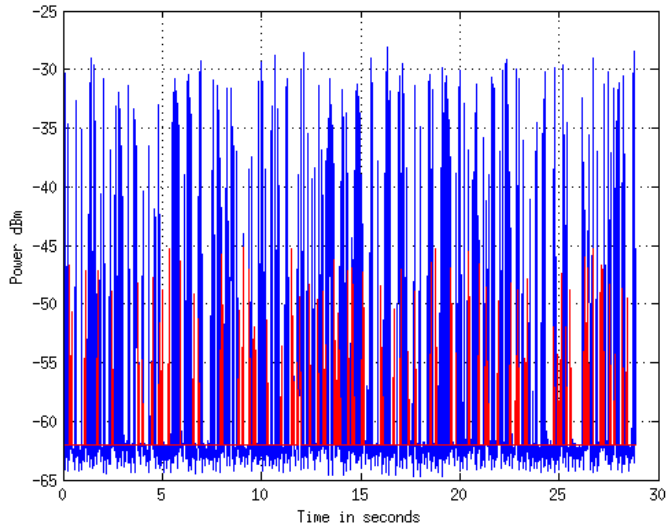


Fig. 18. USRP 2 pos 0, Wi-Fi traffic (Upstream in color red and downstream in color blue)

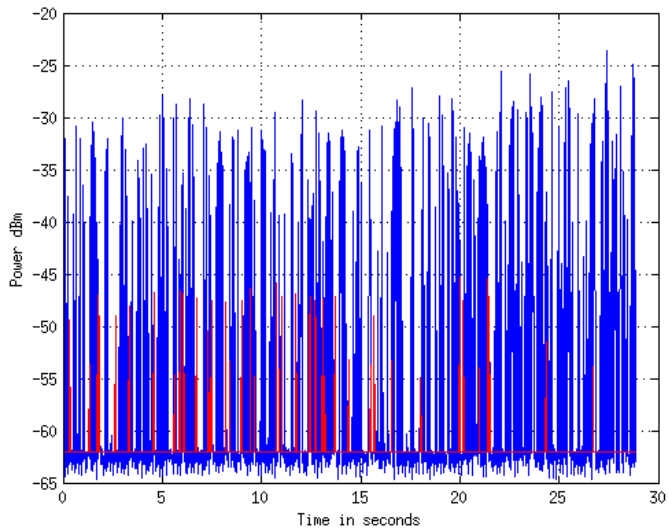


Fig. 19. USRP 2 pos 35, Wi-Fi traffic (Upstream in color red and downstream in color blue)

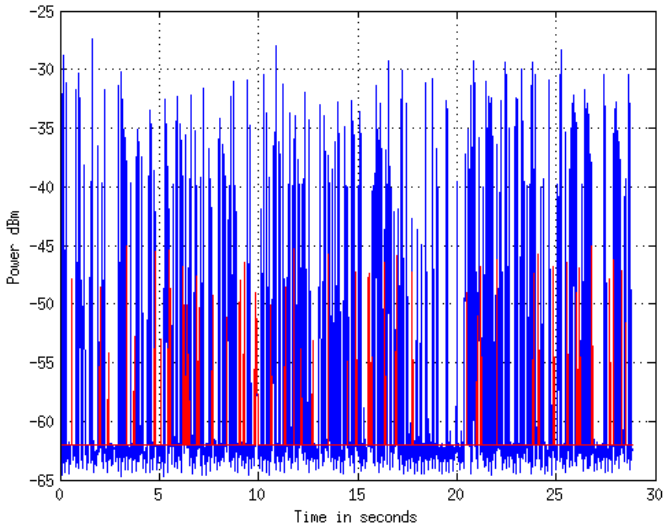


Fig. 20. USRP 2 pos 45, Wi-Fi traffic (Upstream in color red and downstream in color

4.2 DD-WRT wireless broadband routers approach

In order to detect IEEE 802.11 signals with more accuracy an alternative approach using broadband WLAN routers was utilized. This method let us sense the 2.4 – 2.5 GHz spectrum to capture IEEE 802.11 packets for being further analyzed with Wireshark and Matlab. A number of 3 routers are used since this is the minimum number of nodes required for being able to position a terminal.

The routers chosen for this purpose were Linksys WRT54GL (See appendix C). This type of routers is Linux based and are compatible with DD-WRT firmware (See appendix D). DD-WRT is a Linux based alternative OpenSource firmware suitable for a great variety of WLAN routers and embedded systems. The main emphasis lies on providing the easiest possible handling while at the same time supporting a great number of functionalities within the framework of the respective hardware platform used. Compared to the software preinstalled on many WLAN routers, DD-WRT allows a reliable operation with a clearly larger functionality that also fulfills the demands of professional deployment. The main advantages of the DD-WRT firmware are that it permits the installation of certain programs in its hard disk, and that it provides a key functionality to the routers, which is the capability of monitor the IEEE 802.11 network.

To be able to analyze the packets captured using the DD-WRT routers, the packet network analyzers Wireshark and tcpdump are combined. With these tools we can see the packets exchange between the members of a BSS network, so that they can be further processed in Matlab.

4.2.1 Scenario of measurements

The measurements were performed at a typical open space office environment. A map of the scenario is depicted on fig. 21:

meters	0	2.5	5	7.5	10	12.5	15	17.5	20	22.5	25
0	0	5	10	15	20 R1	25	30	35	40	45	50
2.5	1	6	11	16	21	26	31	36	41	46	51
5	2	7	12	17	22	27	32	37	42 R2	47	52
7.5	3	8	13	18	23	28	33	38	43	48	53
10	4	9	14	19	24	29 R3	34	39	44	49	54

Fig. 21. Map of the scenario

The scenario is a rectangular area of 10 x 25 meters that covers most of the space occupied by the office. The grid is set to squares of 2.5 x 2.5 meters, and every spot is numbered from 0 to 54. The origin of coordinates is the spot number 0 (0, 0). Although every spot is separated 2.5 meters from the ones surrounding it, we will just perform measurements in points separated 5 meters. This is suitable distance between two points so that the power levels measured at these points are different enough to make a distinction between the two spots regarding the power levels.

As it can be observed on the map, router 1 (R1) is placed on spot 20, corresponding to coordinates (0 m, 10 m); router 2 (R2) is placed on spot 42, corresponding to coordinates (5 m, 20 m), router 3 (R3) is placed at coordinates (8.95 m, 12.5 m). The height which the routers is placed at is fixed to 1.05 meters. We would have liked to place the routers at a larger height to avoid signal blocking in some parts of the scenario, but it was physically impossible due to the structure of the measurements scenario. The selection of the position of the routers is based on two constraints: good coverage of the area and variability of power levels received at different routers, which enhances indoor positioning algorithms performance. The first premise is fulfilled since the routers give IEEE IEEE 802.11 network access at every spot on the office, and the second is also fulfilled since the distance between routers is large enough to distinguish power levels.

The positions at which the measurements took place were spots 18 (7.5 m, 7.5 m), 26 (2.5 m, 12.5 m), 28 (7.5 m, 12.5 m), 36 (2.5 m, 17.5 m), 38 (7.5 m, 17.5 m), and 45 (2.5 m, 22.5 m).

4.2.2 Detecting IEEE 802.11 signals using DD-WRT wireless routers and packet network analyzers

As mentioned before, DD-WRT firmware gives LinkSys routers the capability of have small programs installed on their hard disk, but also the functionality of monitor every IEEE 802.11 signal in its reach. There are three basic steps in order to detect and analyze IEEE 802.11 packets:

1) The command-line packet network analyzer tcpdump is installed in each router (see Appendix E) and the wireless card each router is set to work in “monitor” mode. This action creates the prism0 interface in the router, which is able to detect every packet transmitted through the IEEE 802.11 air interface.

2) Setting different filters on tcpdump allows discarding unnecessary packets that do not belong to our research. As we are interested on uplink data coming from the mobile phones, we set a specific filter including the MAC addresses of the two mobile devices so that the routers just capture the data that they send. Now we are ready to capture packets.

3) The result of sensing the IEEE 802.11 spectrum with each router is saved in a pcap file in the memory of the router. We send out each file to a computer via LAN, where they are analyzed with another packet network analyzer, Wireshark. With this program we are able to see the data flow that we have captured, with detailed information in every packet related to the IEEE 802.11 protocol. To be able to easily to make calculations and graphs with these files, they are convert to .txt files so that they can be processed with Matlab.

4.2.3 Limitations and assumptions

The routers R1, R2 and R3 form an infrastructure IEEE 802.11 network. The router R1 is connected to the wired network, and the routers R2 and R3 are connected over Ethernet to R1. Thus, all routers work as AP.

When an AP is configured to be connected a specific channel, the prism0 interface that captures the IEEE 802.11 traffic can just detect the packets sent over that channel. Since our purpose is to receive the same packets at each router so that distinction made by power levels received from the same packet at different positions can be done, each router has to be connected to the same channel. This wireless network configuration is not suitable for been deployed for operational purposes, i.e. public networks with thousands of mobile terminals, since there would be channel overlapping leading to interference issues and the performance would decrease significantly. To our practical experience, the performance of the network was not affected by this fact when 4 terminals were connected to it.

However, the routers can be switched to “repeater” mode, so that they are simple sniffers that do not give Internet access, but can perfectly detect IEEE 802.11 signals in the whole IEEE 802.11 b/g bandwidth. In this case, an alternative infrastructure of APs operating in different channels (for instance channels 1, 6 and 11), is assumed to be deployed to give Internet access to users.

4.2.4 Procedure of measurements

Two smartphones (a Samsung and a HTC) were used to test the uplink signals behavior when the mobile phone is connected to our Wi-Fi network, but in idle or inactive state. As mentioned before, the spots where the

measurements took place were 18, 26, 28, 36, 38, and 45. We assumed that, being the phone in Wi-Fi idle mode (corresponding with the Power Saving Mode of IEEE 802.11), the most common places where a person has the phone is in the pocket of a pants/trousers, and also holding it with the hand, with an orientation parallel to the floor. From now on, we will refer to these positions as “pocket” and “standard”. In order to test the influence on RSS by the phone´s antenna, four different directions were added to these positions. In both “pocket” and “standard” positions, directions “0 degrees”, “90 degrees”, “180 degrees” and “270 degrees” were tested to involve the antenna radiation pattern factor to our measurements. In total, 8 measurements were undertaken for each spot in the map and for each mobile phone. The 4 different directions are depicted on fig. 22, together with the map of the scenario.

According to some previous experimental tests, the mobile devices normally send out an average of 2 or 4 packets per minute when operating in Wi-Fi Power Saving Mode, depending on the vendor. Counting the 6 different spots, 2 different positions and 4 different directions, the number of measurements is 48 for each mobile phone. As the measurements procedure requires active intervention by a person to recreate a real scenario, and since this person has to hold one mobile phone with the hand, having the other phone in the pocket at each corresponding spot, position and direction, we assumed that 5 minutes were suitable to received enough packets to make averaging over time, but also to not prolong considerably the duration of the measurements, which was already 240 minutes (4 hours) without counting time between measurements, etc.

In order to test the reproducibility and repeatability of the measurements (important factor for the fingerprint technique), we performed the same type of measurements a different day. An additional test was undertaken to examine the possible difference in power levels between the uplink packets sent by the phones when operating in PSM or when sending out data traffic. For this purpose, the same measurement procedure (two smartphones, different spots, positions and directions) was utilized to test the RSS at each router when accessing a website (Facebook, for being one of the most visited), watching a video, and making a Skype call. As in this mode the mobile phone transmits larger amount of packets, shorter durations of the measurements are needed.

meters	0	2.5	5	7.5	10	12.5	15	17.5	20	22.5	25
0	0	5	10	15	20 R1	25	30	35	40	45	50
2.5	1	6	11	16	21	26	31	36	41	46	51
5	2	7	12	17	22	27	32	37	42 R2	47	52
7.5	3	8	13	18	23	28	33	38	43	48	53
10	4	9	14	19	24	29 R3	34	39	44	49	54

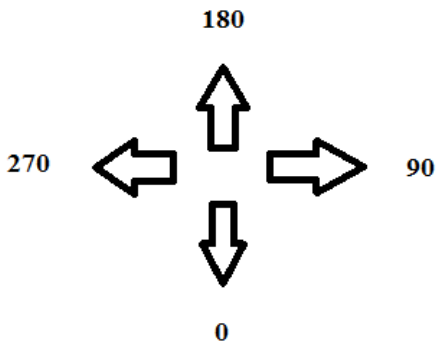


Fig. 22. Orientations of the mobile phone used in the measurements

Routers R1, R2 and R3 operate in the same SSID and in channel 1, which center frequency is 4.12 GHz. The beacon interval is 10 milliseconds and DTIM is set to the default value, 3. As it will be demonstrated with the experimental results, these values do not affect the periodicity of the uplink signals.

CHAPTER 5

5 Results

5.1 USRP results

5.1.1 Distinction between upstream and downstream

Based on the measurements done with the USRP's, it was observed that trying to separate between upstream and downstream based on power levels was quite challenging as IEEE 802.11 share the whole bandwidth for either upstream or downstream not using FDMA . The physical layer is the same for both streams and IEEE 802.11 does not make any difference using different modulations, or techniques for a specific stream direction.

In addition, if it is desired to achieve the distinction based on power levels, it could depend on how the Wi-Fi network is designed and where the USRP's are placed. Moreover, multipath effects could have the effect that some signals will be affected by some fading dips for example and the power levels do not fall within our thresholds. Furthermore, in some cases some upstream that could not fall within our thresholds could be considered as downstream.

One possible solution to the mentioned above, as USRP works with blocks done in C++ and the linking between blocks is done with python, we could develop a receiver for IEEE 802.11 and decode the information required for the distinction.

The first two bytes of the MAC packet structure are composed by the frame control. The frame control contains two flags which indicate if the traffic is generated either by the access point or station.

We realized that decoding IEEE 802.11 packets could be done with simple

routers, hence, to develop an IEEE 802.11 receiver would be time consuming, and it is out of the scope of our thesis work and timelines, furthermore, about implementation cost would be reduced considerably with just simple routers. So we decided to use routers instead of USRP.

5.1.2 Uniqueness of the signal

It is important to distinguish between upstream and downstream signals on Wi-Fi traffic. Once, it can be done the distinction between upstream and downstream, it is important to make unique each signal.

For this task, it was decided to use the MAC address as it is unique per hardware. The MAC address does not change as the IP address. The MAC address can be obtained by routers.

5.2 Results of measurements with IEEE 802.11 wireless routers

5.2.1 Idle mode experimental behavior

One of the most important issues regarding the IEEE 802.11 uplink traffic is the traffic pattern during PSM. As we are interested in locating a mobile device by detecting the packets that it transmits, these packets have to be transmitted within a time gap that has to be short enough to track the mobility pattern of a person accurately. The figures presented below show the PSM traffic pattern during the 5 minutes of a measurement for the two different mobile devices.

From fig. 23 is observed that the uplink traffic pattern of the HTC is characterized by the transmission of a packet every 15 seconds. The bit PWR MGT is set to “1”, which means that the mobile device is operating in Power Saving Mode. As the information transmitted is “Data”, we deduce that the phone is responding to a DTIM beacon sent by the AP with a PS-Poll frame containing data. We should also see the ACK packets sent by the mobile device to the AP, but the MAC source address of these packets is not specified so the Wireshark filter does not detect the ACK packets.

Time	MAC source address	Information	Power
0.000000	Htc_be:bb:db	Data, SN=4038, FN=0, Flags=.p.P...T	-48
15.368606	Htc_be:bb:db	Data, SN=4039, FN=0, Flags=.p.P...T	-47
30.733023	Htc_be:bb:db	Data, SN=4040, FN=0, Flags=.p.PR..T	-46
46.100737	Htc_be:bb:db	Data, SN=4041, FN=0, Flags=.p.P...T	-47
61.467408	Htc_be:bb:db	Data, SN=4042, FN=0, Flags=.p.PR..T	-46
76.832880	Htc_be:bb:db	Data, SN=4043, FN=0, Flags=.p.P...T	-47
92.197159	Htc_be:bb:db	Data, SN=4044, FN=0, Flags=.p.PR..T	-48
107.562426	Htc_be:bb:db	Data, SN=4045, FN=0, Flags=.p.PR..T	-47
122.926231	Htc_be:bb:db	Data, SN=4046, FN=0, Flags=.p.PR..T	-48
122.926381	Htc_be:bb:db	Data, SN=4046, FN=0, Flags=.p.PR..T	-48

```
▼ IEEE 802.11 Data, Flags: .p.P...T
    Type/Subtype: Data (0x20)
  ▼ Frame Control: 0x5108 (Normal)
      Version: 0
      Type: Data frame (2)
      Subtype: 0
    ▼ Flags: 0x51
        .... ..01 = DS status: Frame from STA to DS via an AP (To DS: 1 From DS: 0) (0x01)
        .... .0.. = More Fragments: This is the last fragment
        .... 0... = Retry: Frame is not being retransmitted
        ...1 .... = PWR MGT: STA will go to sleep
        ..0. .... = More Data: No data buffered
        .1.. .... = Protected flag: Data is protected
        0... .... = Order flag: Not strictly ordered
    Duration: 40
    BSS Id: Cisco-Li_4c:78:62 (68:7f:74:4c:78:62)
    Source address: Htc_be:bb:db (90:21:55:be:bb:db)
    Destination address: Broadcast (ff:ff:ff:ff:ff:ff)
```

Fig. 23. HTC uplink traffic pattern 1

Time	MAC source address	MAC destination address	Information
0.000000	Htc_be:bb:db	Cisco-Li_1a:6d:0d	Data, SN=1981, FN=0, Flags=.p.P...T
48.093416	Htc_be:bb:db	Cisco-Li_1a:6d:0f	Null function (No data), SN=1982, FN=0, Flags=...PR..T
55.801818	Htc_be:bb:db	Cisco-Li_1a:6d:0d	Data, SN=1983, FN=0, Flags=.p.PR..T
60.791915	Htc_be:bb:db	Cisco-Li_1a:6d:0d	Data, SN=1984, FN=0, Flags=.p.PR..T
109.538618	Htc_be:bb:db	Cisco-Li_1a:6d:0f	Null function (No data), SN=1985, FN=0, Flags=...PR..T
116.599787	Htc_be:bb:db	Cisco-Li_1a:6d:0d	Data, SN=1986, FN=0, Flags=.p.PR..T
121.603430	Htc_be:bb:db	Cisco-Li_1a:6d:0d	Data, SN=1987, FN=0, Flags=.p.P...T
170.985443	Htc_be:bb:db	Cisco-Li_1a:6d:0f	Null function (No data), SN=1988, FN=0, Flags=...PR..T
177.397468	Htc_be:bb:db	Cisco-Li_1a:6d:0d	Data, SN=1989, FN=0, Flags=.p.PR..T
177.397905	Htc_be:bb:db	Cisco-Li_1a:6d:0d	Data, SN=1989, FN=0, Flags=.p.PR..T
177.398994	Htc_be:bb:db	Cisco-Li_1a:6d:0d	Data, SN=1989, FN=0, Flags=.p.PR..T
182.397657	Htc_be:bb:db	Cisco-Li_1a:6d:0d	Data, SN=1990, FN=0, Flags=.p.P...T
182.430792	Htc_be:bb:db	Cisco-Li_1a:6d:0d	Data, SN=1991, FN=0, Flags=.p.P...T
232.431708	Htc_be:bb:db	Cisco-Li_1a:6d:0f	Null function (No data), SN=1992, FN=0, Flags=...P...T
239.195021	Htc_be:bb:db	Cisco-Li_1a:6d:0d	Data, SN=1993, FN=0, Flags=.p.P...T
244.195497	Htc_be:bb:db	Cisco-Li_1a:6d:0d	Data, SN=1994, FN=0, Flags=.p.PR..T
244.196102	Htc_be:bb:db	Cisco-Li_1a:6d:0d	Data, SN=1994, FN=0, Flags=.p.PR..T
244.197026	Htc_be:bb:db	Cisco-Li_1a:6d:0d	Data, SN=1994, FN=0, Flags=.p.PR..T
244.201065	Htc_be:bb:db	Cisco-Li_1a:6d:0d	Data, SN=1994, FN=0, Flags=.p.PR..T
244.230902	Htc_be:bb:db	Cisco-Li_1a:6d:0d	Data, SN=1995, FN=0, Flags=.p.P...T
293.872112	Htc_be:bb:db	Cisco-Li_1a:6d:0f	Null function (No data), SN=1996, FN=0, Flags=...PR..T
300.133868	Htc_be:bb:db	Cisco-Li_1a:6d:0d	Data, SN=1997, FN=0, Flags=.p.PR..T
300.134443	Htc_be:bb:db	Cisco-Li_1a:6d:0d	Data, SN=1997, FN=0, Flags=.p.PR..T

```
▸ Frame 1: 216 bytes on wire (1728 bits), 216 bytes captured (1728 bits)
▸ Prism capture header
▸ IEEE 802.11 Data, Flags: .p.P...T
▸ Data (40 bytes)
```

Fig. 24. HTC uplink traffic pattern 2

From fig. 24 is observed that the uplink traffic pattern of the HTC in this case is different. Although it is not displayed, all the packets have the PWR MGT bit set to “1”, so the mobile device is operating in Power Saving Mode. In this case the traffic pattern of the HTS is characterized by the following sequence:

Null function packet – *around 7 seconds* – Data packet – *around 5 seconds* – Data packet – *around 49 seconds* – Null function packet (and the sequence starts again).

When there are more than two data packets in a row, is due to failed transmissions of a data packet, so it is retransmitted until the packet reaches its destination (the retransmitted packets have the same sequence number). From the figure it can be deduced that every 60 seconds (approximately), the mobile device responds with a Null function packet to a previous DTIM beacon sent by an AP, which means that the device does not have any data packet to transmit. In between, and also with similar periodicity, two data packets are transmitted by the mobile phone.

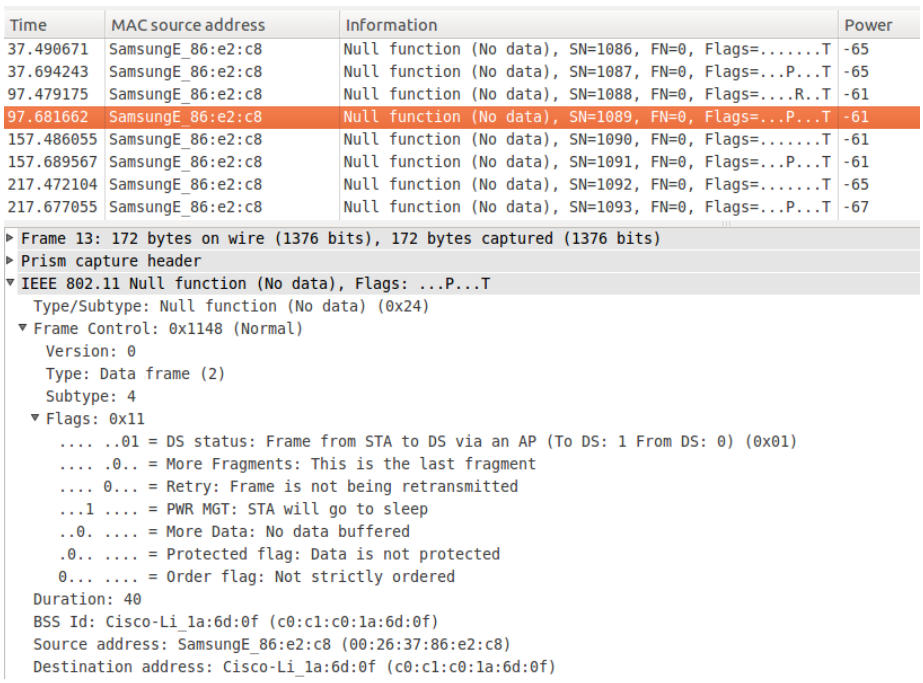


Fig. 25. Samsung uplink traffic pattern

The fig. 25 shows the IEEE 802.11 uplink traffic pattern of the Samsung mobile phone. As the HTC, is operating in Power Saving Mode, but in this case it transmits two consecutive Null function packets every 60 seconds (approximately). The first packet informs the AP that the mobile device does not have any information to transmit, and the second packet is also Null function type, but it also informs the AP that it will go to sleep. After 60 seconds, the mobile device will repeat the same procedure. This behavior is always the same, contrary to the HTC case.

As a conclusion we observe that there is a significant difference of behavior between different vendors. The reason might be that each vendor follows different power saving policies by setting different timers when operating in IEEE 802.11 PSM. In any case both mobile devices send out enough packets to be detected for the purpose of tracking mobility patterns. Regarding the different traffic pattern that the HTC experiments, is worth to say that the configuration of the network and the number of STAs connected to it was exactly the same in all tests, so the reason for this behavior might be the amount of battery that the mobile phone had at different moments.

5.2.2 Variability analysis

Variability in power levels received at the different wireless routers is one of the key factors of a successful indoor positioning technique. If there is a remarkable difference in power levels received by each router at the different spots, the performance of the positioning algorithm is enhanced since the different positions computed depend on these power levels. In the figures 26, 27, 28 and 29, it is demonstrated that the variability constraint is fulfilled in our measurements:

The fig. 25, 26, 27, and 28 show the average power over 5 minutes received at each router, at different spots, at different positions, at different orientations and for both mobile phone models. It can be noticed that there is a great variability of power levels depending on the router and the orientation of the mobile phone. This fact reflects the influence of the antenna radiation pattern on the measurements, and it adds more diversity to the measurements. The RSS at each router also varies depending on the distance to the different spots, according to a radio propagation scenario.

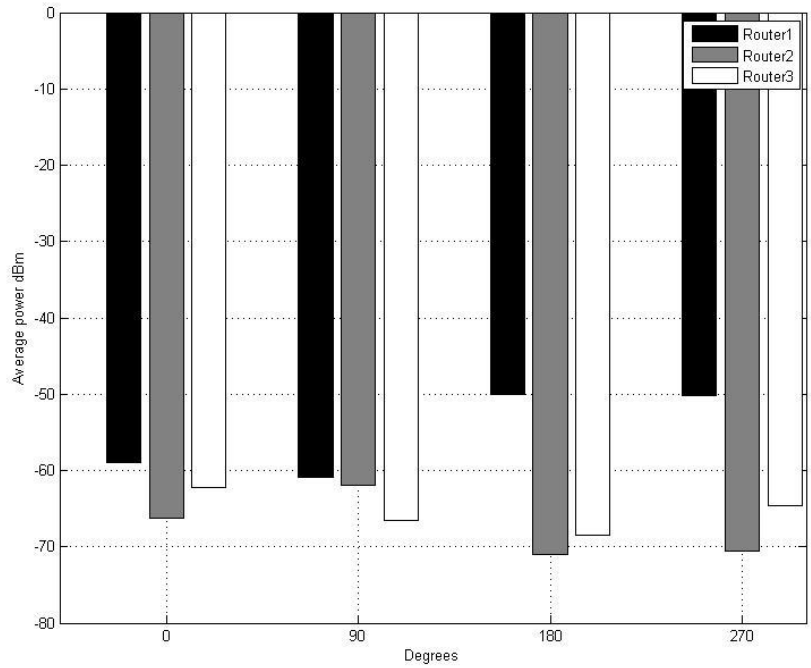


Fig. 26. Average power-Samsung-spot 26-standard position

TABLE 3. POWER STANDARD DEVIATION (dB)-SAMSUNG-SPOT 26-STANDARD POSITION

	0 degrees	90 degrees	180 degrees	270 degrees
Router 1	0,700	1,467	0	1,994
Router 2	1,982	1,414	0,816	1,378
Router 3	1,658	2,759	0,849	0,699

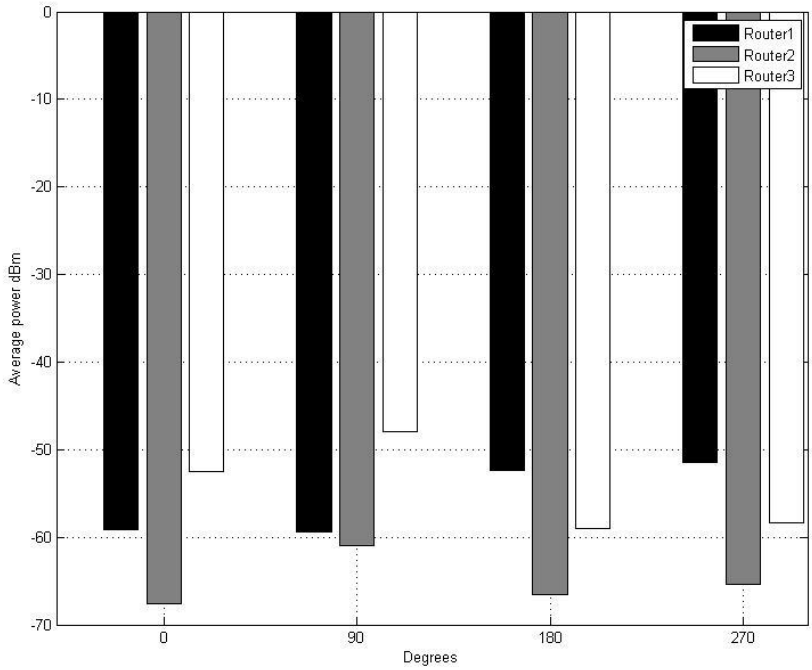


Fig. 27. Average power-Samsung-spot 28-pocket position

TABLE 4. POWER STANDARD DEVIATION (dB)-SAMSUNG-SPOT 28-POCKET POSITION

	0 degrees	90 degrees	180 degrees	270 degrees
Router 1	1,911	1,173	2,949	2,118
Router 2	0,520	1,513	0,967	0,516
Router 3	0,527	0,816	0,953	0,699

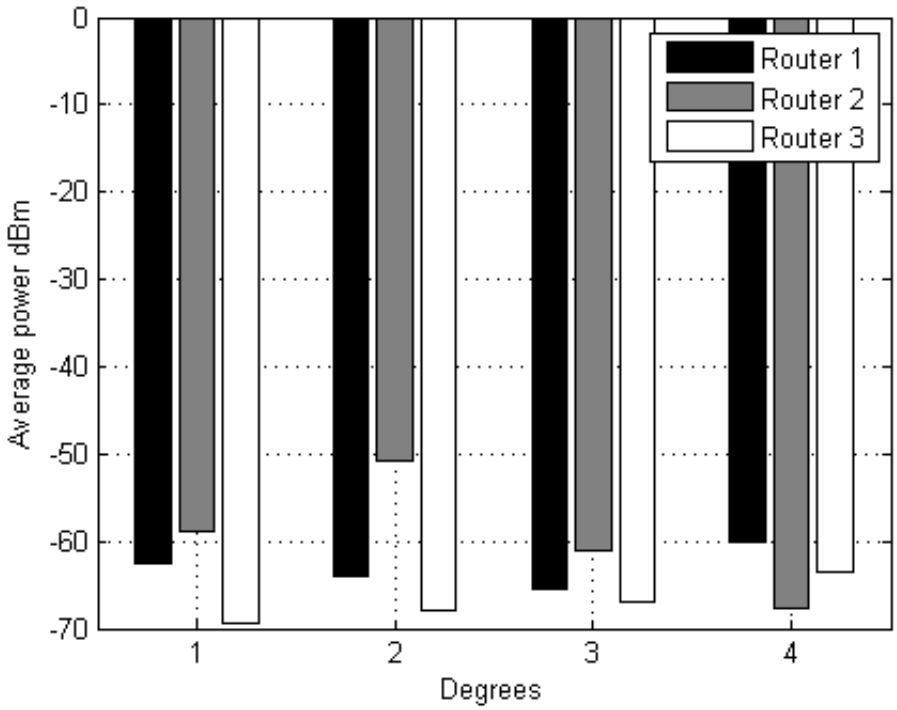


Fig. 28. Average power-HTC-spot 36-pocket position

TABLE 5. POWER STANDARD DEVIATION (dB)-HTC-SPOT 36-POCKET POSITION

	0 degrees	**90 degrees**	**180 degrees**	**270 degrees**
Router 1	1,053	0,848	1,761	1,671
Router 2	0,977	0,455	0,936	1,329
Router 3	1,643	1,759	1,712	1,667

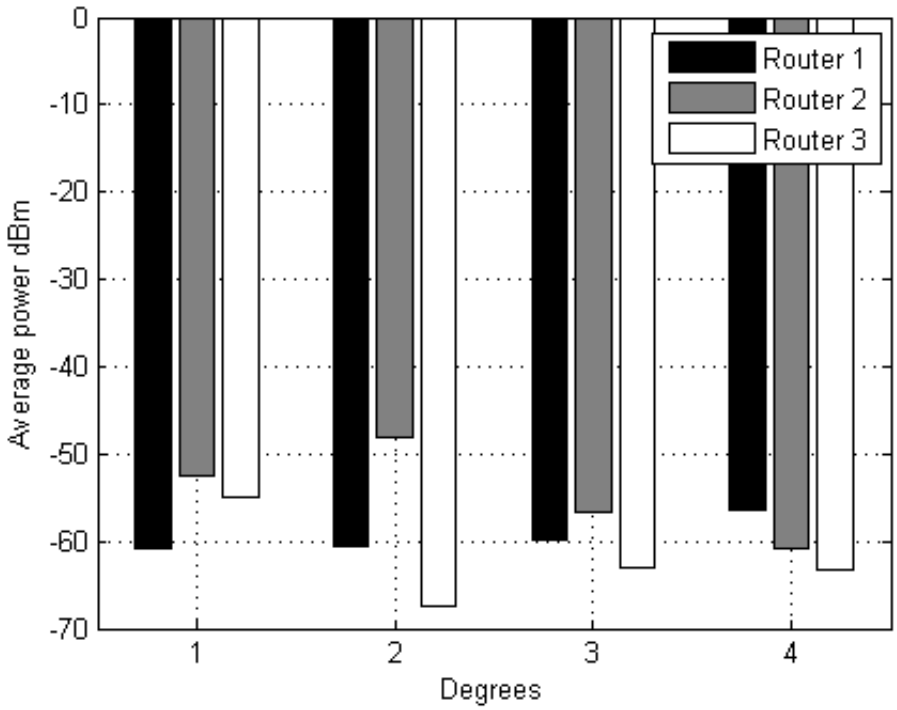


Fig. 29. Average power-HTC-spot 38-standard position

TABLE 6. POWER STANDARD DEVIATION (dB)-HTC-SPOT 38-STANDARD POSITION

	0 degrees	90 degrees	180 degrees	270 degrees
Router 1	1,424	2,123	1,820	2,539
Router 2	1,306	1,268	1,647	1,781
Router 3	2,268	1,658	1,197	1,279

The tables 3, 4, 5, and 6 represent the corresponding standard deviation of power levels, due to averaging over 5 minutes. It can be observed that this deviation is in the range 1- 2 dB, which means that the time chosen for the measurements is good enough to make an accurate average of power levels.

The received power levels at each router from each spot are averaged over the 4 different directions, to give an overall idea of the variability of RSS omitting the antenna factor. Figure 30, 31, 32 and 33 show this behavior.

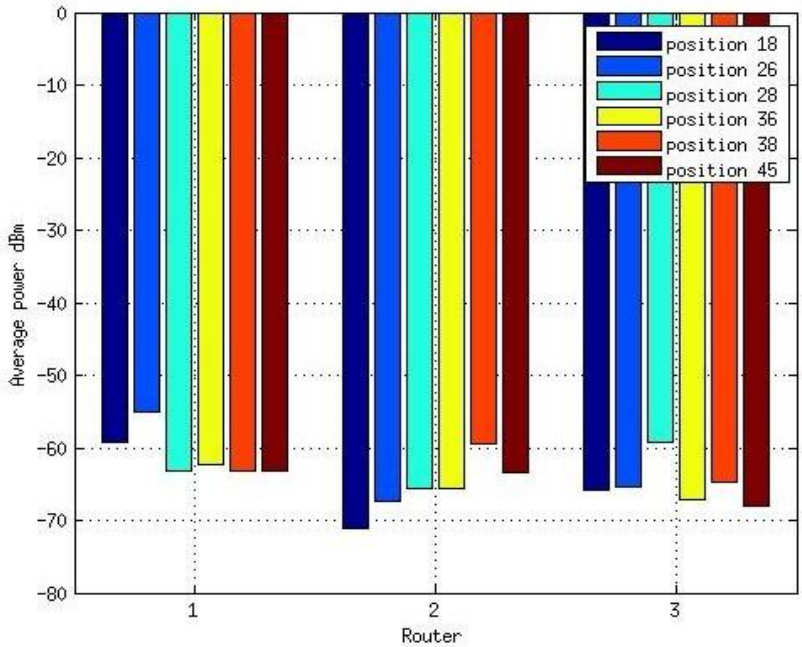


Fig. 30. Average power-Samsung-all spots-standard position

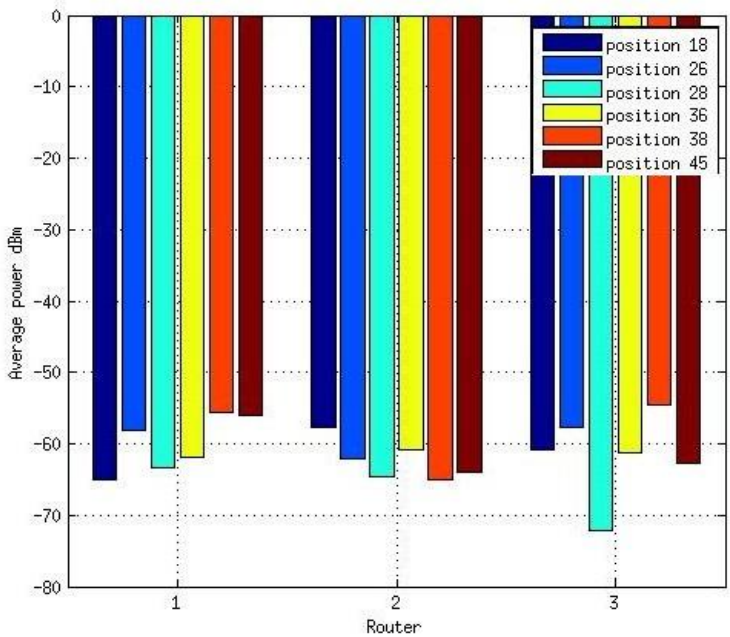


Fig. 31. Average power-Samsung-all spots-pocket position

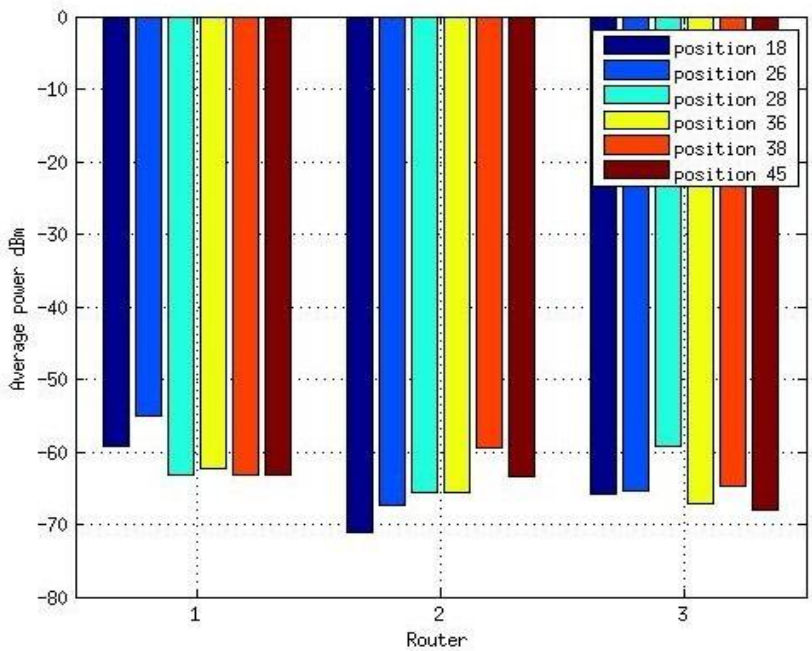


Fig. 32. Average power-HTC-all spots-standard position

In these graphs the great variability of power levels is confirmed. The RSSs at each router differs considerably between them for both standard and pocket positions.

There is also a slight variability between the power levels received when the phone is in pocket or standard position. This is normal since they have different heights.

Regarding the difference of power levels received from the two Smartphones, the total average power received from the HTC is -60.09 dBm, while the average power received from the Samsung is -62.55 dBm. As the IEEE 802.11 standard jus sets a maximum transmitted power of the mobile devices, different vendors can vary the transmitted power value to fit the features of the device according to the different power saving policies that they follow. From our point of view, the vendor of the mobile device shall not be an important factor to consider when it comes to transmitted power levels.

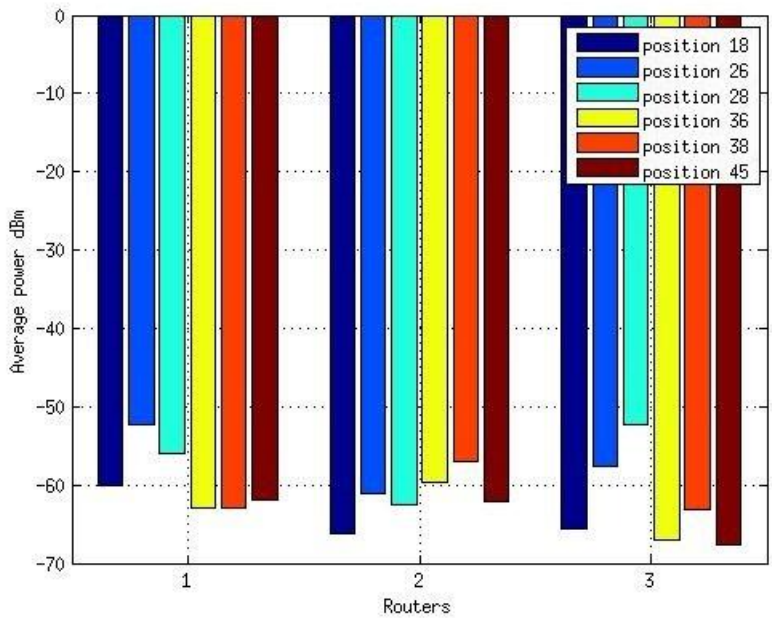


Fig. 33. Average power-HTC-all spots-pocket position

The main conclusion obtained from the analysis of the variability of received power levels is that these RSS are different enough for being used within any indoor positioning technique.

5.2.3 Reproducibility analysis

Reproducibility is another important requirement for a successful indoor positioning method. The received power levels at any time at the routers from the same device must be similar so that there is no dependence of when the measurements are performed. Also, there must be no dependence of the type of traffic that the mobile device is transmitting, since we assume (according to the information provided by the IEEE 802.11 standards) that the power transmitted by a mobile phone when transmitting IEEE 802.11 packets does not depend of the type of traffic. Obviously, the mobile device transmits more packets when accessing a website, downstreaming a video or making a Skype call, but this fact shall not affect the transmitted power in every packet.

In the figures depicted below, relevant comparisons between same measurements performed at different days are presented. In addition, a comparison between the power levels received in Power Saving Mode (idle mode) and data traffic is also depicted.

In order to reduce time employed in the measurements, RSS at each router with the phone in standard position at spots 26, 28, 36 and 38 were tested.

On fig. 34, fig. 35, fig. 36, and fig. 37 four graphs compare the same measurements performed under the same conditions but at different days. As it can be observed, the power levels patterns are pretty similar, confirming the reproducibility of the measurements over time.

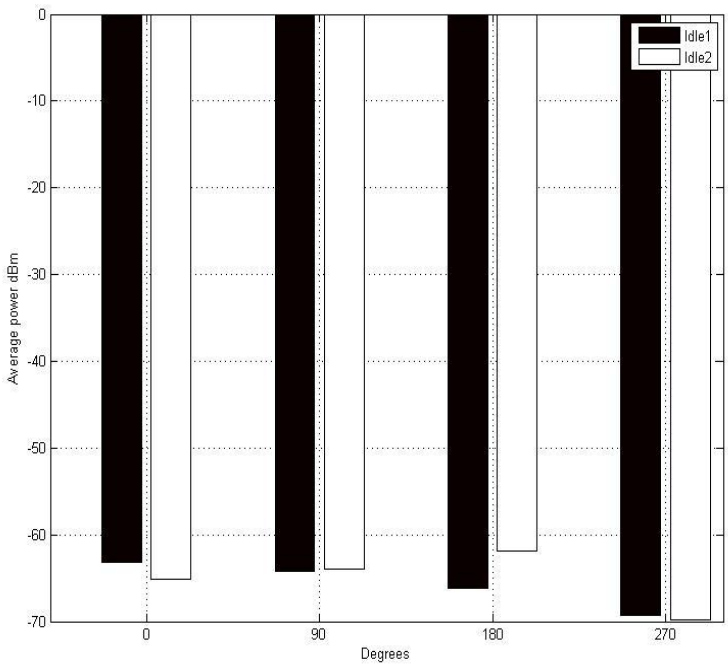


Fig. 34. Average power levels at different days-idle mode-router 2-Samsung-spot 36

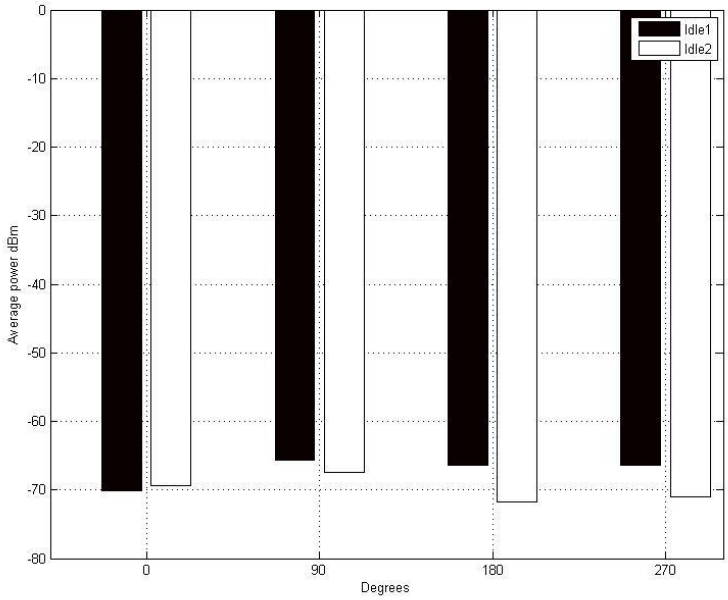


Fig. 35. Average power levels at different days-idle mode-router 3-Samsung-spot 36

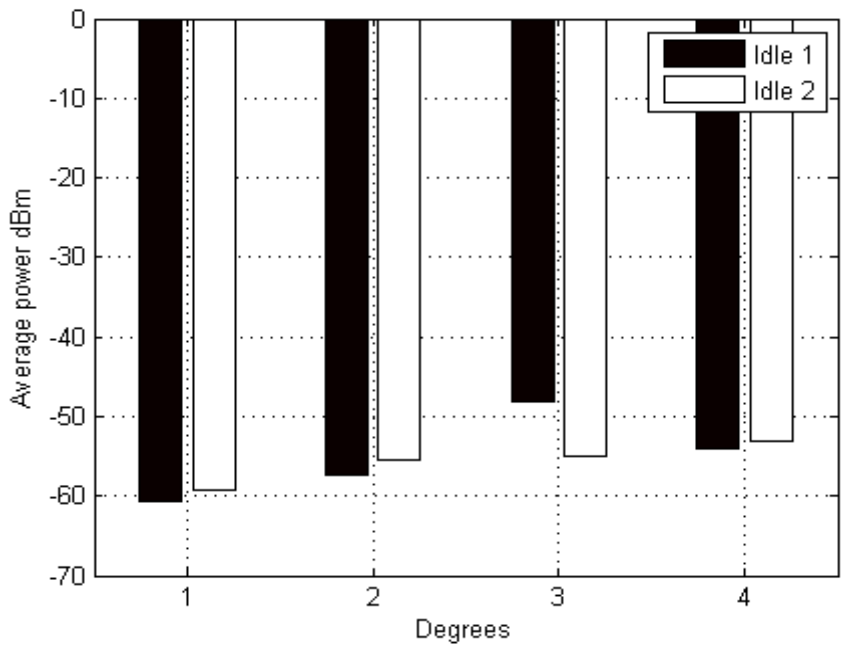


Fig. 36. Average power levels at different days-idle mode-router 1-HTCspot 28

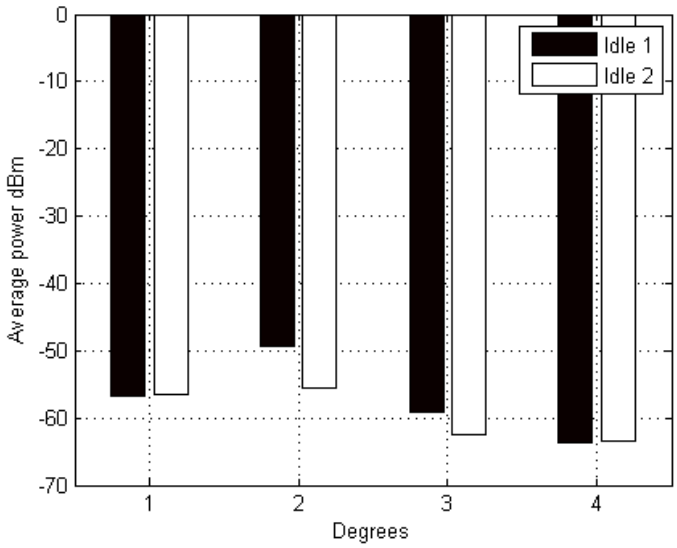


Fig. 37. Average power levels at different days-idle mode-router 2-HTC-spot 28

Some examples of the comparison of RSS with different data traffic are presented next:

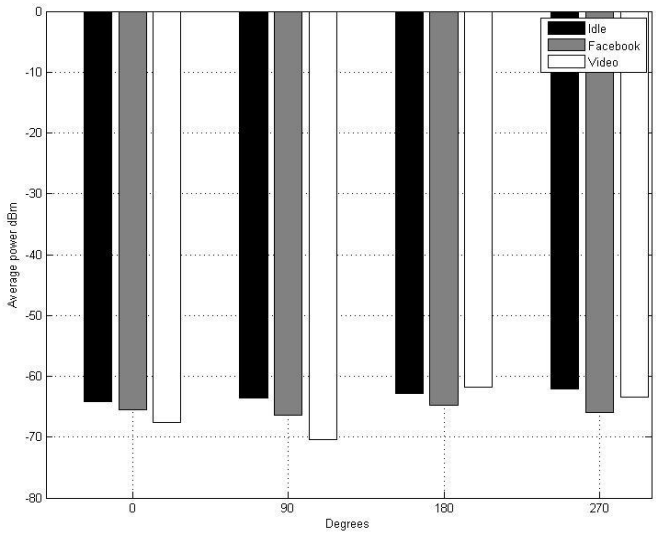


Fig. 38. Average power levels at different days-idle vs traffic-router 1-Samsung-spot 38

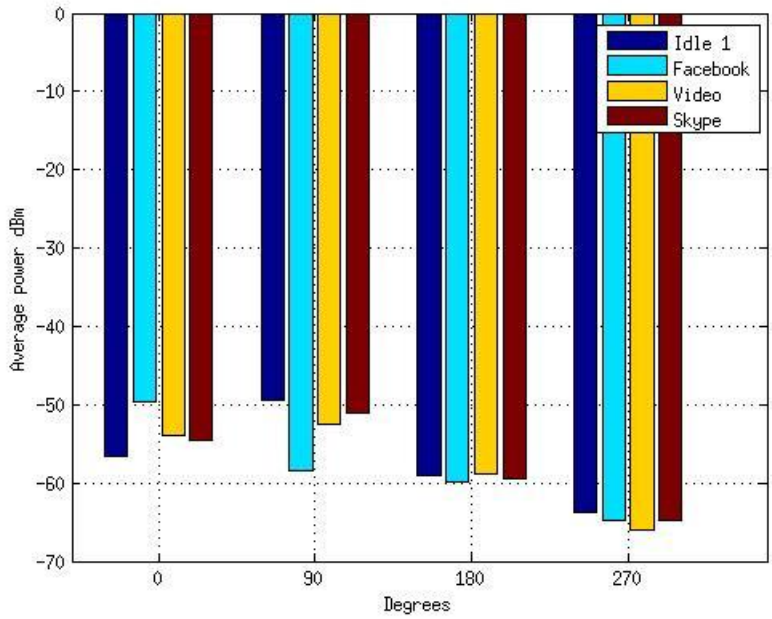


Fig. 39. Average power levels at different days-idle vs traffic-router 2-Samsung-spot 26

5.2.4 Practical application using trilateration

The measurements and trilateration technique were applied in an office environment. In a normal trilateration scenario, one device receives power levels from three transmitters, and applying radio propagation models the distances between each transmitter and the receiver are calculated. Using the geometry of the three circles with radii equal to the distances calculated previously, the position can be estimated. In our research we have employed the most basic trilateration algorithm, given by the functions (4), (5), and (6).

$$(x-x1)^2+(y-y1)^2+(z-z1)^2=d1^2 \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (4)$$

$$(x-x2)^2+(y-y2)^2+(z-z2)^2=d2^2 \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (5)$$

$$(x-x3)^2+(y-y3)^2+(z-z3)^2=d3^2 \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (6)$$

The coordinates of router i (i =1, 2, 3) are known (x_i, y_i, z_i) and the distances between each router and the corresponding spot (d_i) are computed using a radio propagation model for indoor environments. Solving the

system of equation for each spot, the coordinates x, y, z (although z coordinated is not relevant for our purpose and it will not be computed) corresponding to the estimated position of the spot are obtained. In our scenario, although the role of the participants is reverse (one single transmitter, three receivers), the geometry constraints are the same as in a normal trilateration scenario. The figure below shows a sketch of the scenario

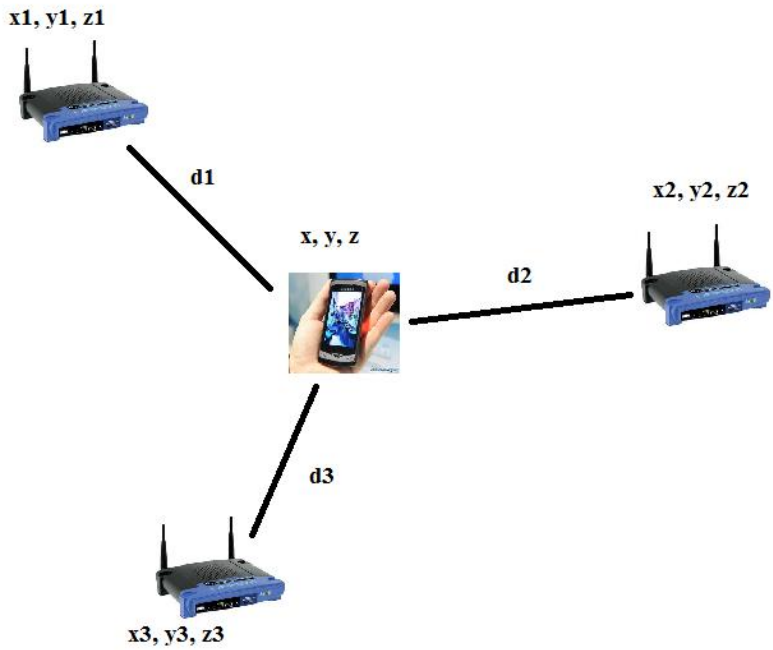


Fig. 40. Reverse trilateration scenario

The system of equations with numerical values of x_i, y_i, z_i is:

$$(x-0)^2+(y-10)^2+(z-1.05)^2=d1^2 \quad \text{.....(7)}$$

$$(x-5)^2+(y-20)^2+(z-1.05)^2=d2^2 \quad \text{.....(8)}$$

$$(x-8.95)^2+(y-12.5)^2+(z-1.05)^2=d3^2 \quad \text{.....(9)}$$

The radio propagation model employed is the ITU site-general model for path loss prediction in indoor environments [20]:

$L_{TOTAL} = 20log_{10}f + Nlog_{10}d + Lf(n) - 28$.....(10)

where f is the frequency in MHz, N is the power decay index, d is the distance in meters (d > 1 meter), Lf(n) is the floor loss propagation factor and n is the number of floors (n > 1). As in our case n=1, Lf (n) is equal to 0. Thus, the radio propagation model for 1 floor is as follows:

$L_{TOTAL} = 20log_{10}f + Nlog_{10}d - 28$ ……(11)

The path loss can be also expressed as the difference in dB between the transmitted and the received power:

$L_{TOTAL} = P_{TX} - P_{RX}$ ……(12)

By merging equations 11 and 12 we can express the received power P_{TX} with the following formula:

$P_{RX} = P_{TX} - 20log_{10}f - Nlog_{10}d + 28$ ……*(13)*

For each measurement, we obtain three levels power levels at each router from a distance d to the phone, which is located at a specific spot in the map:

$P_{RX1} = P_{TX} - 20log_{10}f - Nlog_{10}d_1 + 28$……*(14)*
$P_{RX2} = P_{TX} - 20log_{10}f - Nlog_{10}d_2 + 28$……*(15)*
$P_{RX3} = P_{TX} - 20log_{10}f - Nlog_{10}d_3 + 28$……*(16)*

As P_{TX} and f are fixed values common to equations (14),(15) and (16), we can create a system of 3 equations with 4 unknown factors by combining the equations as follows:

$P_{RX1} - P_{RX2} = Nlog_{10}(d_2/ d_1)$……*(17)*
$P_{RX2} - P_{RX3} = Nlog_{10}(d_3/ d_2)$……*(18)*
$P_{RX3} - P_{RX1} = Nlog_{10}(d_1/ d_3)$……*(19)*

From equations (4), (5) and (6) we know that the distance is a function of the coordinates x, y and z. As we are not interested in the coordinate z, we can write d_1, d_2 and d_3 as a function of x and y, and include these

expressions in the equations (17), (18) and (19), creating the following system:

$$P_{RX1} - P_{RX2} = \frac{1}{2} N log_{10} \frac{(x-5)^2 + (y-20)^2}{x^2 + (y-10)^2} \quad \text{.....(20)}$$

$$P_{RX2} - P_{RX3} = \frac{1}{2} N log_{10} \frac{(x-8.95)^2 + (y-12.5)^2}{(x-5)^2 + (y-20)^2} \quad \text{.....(21)}$$

$$P_{RX3} - P_{RX1} = \frac{1}{2} N log_{10} \frac{x^2 + (y-10)^2}{(x-8.95)^2 + (y-12.5)^2} \quad \text{.....(22)}$$

As a result, in equations (20), (21) and (22) we have a system of three equations with three unknown factor which are x, y and N.

For spots 26, 28, 36 and 38 in the map we collected 40 vectors of received powers at the routers (P_{RX1}, P_{RX2}, P_{RX3}). These power levels are formed by 4 sectors that contain 10 values representing each of the 4 directions measured at each spot (0, 90, 180 and 270) degrees). Spots 18 and 45 are not considered since they are not fully covered by the routers.

The computation of x, y and N was performed using the function *fsolve* of Matlab, that uses the algorithm trust region dogleg to solve non-linear equations. The algorithm needs a starting point or a "guess" of the unknown factors to solve the equations. In a real scenario, fingerprinting could be used to give the trust region dogleg algorithm the starting point, comparing the received vector of power levels with a list of fingerprints linked to each spot. The fingerprint that matches better the received vector of power levels would give the x and y coordinates to be used as the starting point for the trust region dogleg algorithm. As we would have just a few values in our vector of power levels (in our case, 3), fingerprinting would not be effective by itself. Thus, the combination of these two techniques may result in a significant enhancement of the positioning technique.

Figs. 41 and 42 show a comparison between the real positions (marked with squares) and the estimated positions (marked with asterisks). Each color represents one spot.

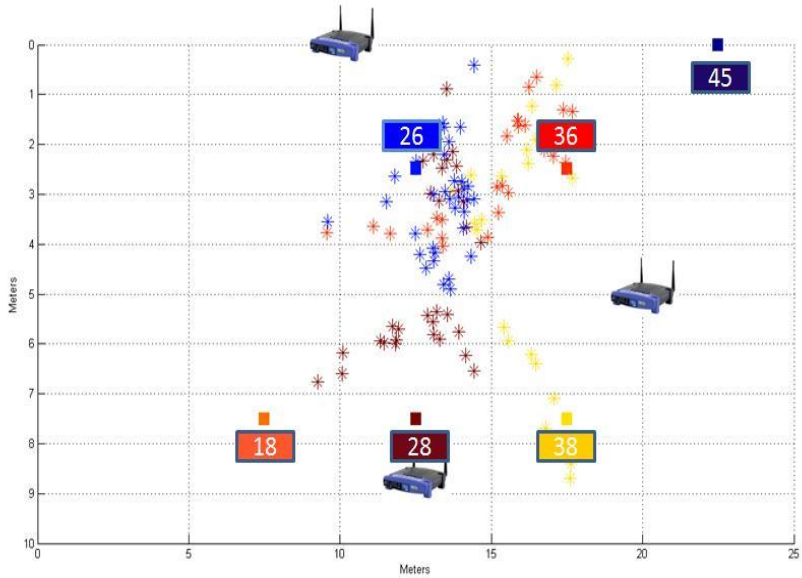


Fig. 41. Estimated positions vs real position of each spot (HTC)

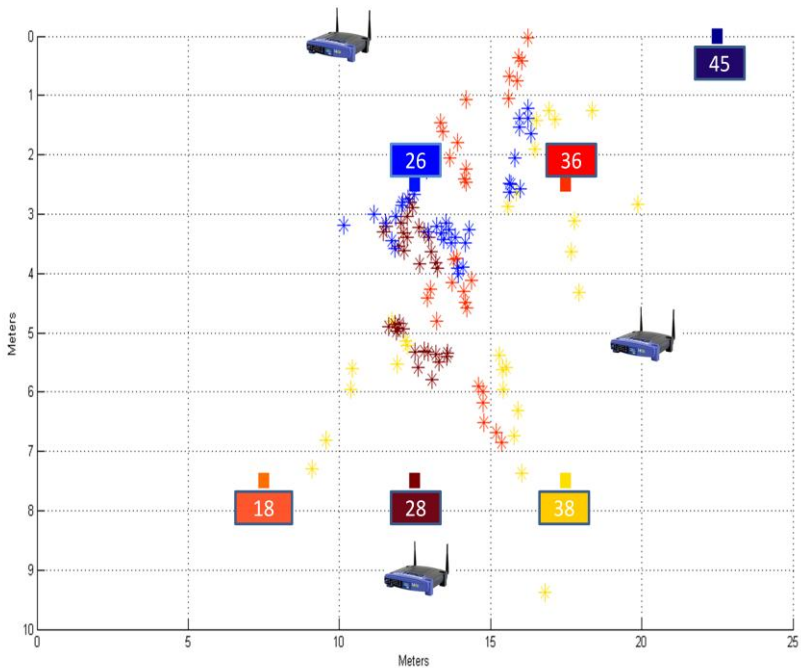


Fig 42. Estimated positions vs real position of each spot (Samsung)

The average distance deviation is 2.2626 meters, which represents a great accuracy for an indoor positioning system. However, we believe that the accuracy would increase with a larger number of receivers covering the entire scenario, which at the same time would enable to position the phone at more spots.

CHAPTER 6

6 Conclusions and future work

The main goal during this thesis project was to answer the three questions presented in the abstract, which were the base of this project work.

First question: How often the mobile device sends out signals during idle mode state for the different radio sources?

The analysis showed that GSM and UMTS do not present regular and continuously signal activity during the idle mode (from 6 minutes to 25.5 hours). On the other hand, IEEE 802.11 presents more regular signal activity during the sleeping mode state (an average of three packets per minute) so, IEEE 802.11 can be used for tracking mobile phones in indoor environments.

In the future work, it would be interesting to figure out the way to increase the regularity of signal activity during idle mode state by changing the channel on the routers. There should be a balance between battery life and regular signal activity.

Second question: Are these uplink signals feasible for a general indoor positioning technique?

Through our analysis, we concluded that the data acquired using DD-WRT routers can be used for any indoor positioning technique, since the requirements of variability and reproducibility are fulfilled. In the practical analysis of trilateration, positive results were obtained as a starting point for a deeper research in this model. The combination of fingerprinting and trilateration is an attractive approach, and thus further

In the future work, it would be necessary to test the combination fingerprinting and trilateration as it was suggested in this master thesis. Combining such techniques would be a powerful and attractive approach for indoor positioning. Furthermore, it would be definitely useful to

increase the number of routers/receivers and to simulate more advanced algorithms to increase precision of the positioning technique. In addition, radio environment analyzers like the Ekahau HeatMapper would give a better estimation of the coverage of the receivers so that they can be placed in an effective way.

Third question: Is there a way to find uniqueness between terminals?

We concluded that using the MAC address on IEEE 802.11 networks is feasible to have uniqueness between different users.

To sum it up: IEEE 802.11 uplink signals can be used for indoor tracking. We have been able to separate upstream and downstream traffic with simple routers and DD-WRT software installed on them. Furthermore, there is no need to have a software program installed on the mobile phone. We have reduced user intervention. In addition, trilateration has shown fairly positive results.

On the other hand, routers are a lot cheaper than USRP devices. When it comes to implementation cost, it will require less investment on the hardware.

One main disadvantage is that it is required that the Wi-Fi transceiver is turned on. One solution might be to provide free internet in the desired area to cover.

Another disadvantage is that in some mobile phones the default Wi-Fi sleep policy configuration is set to last 15 min or until the screen is off. It means that after 15 min or when the screen is off, the Wi-Fi transceiver will be switched off. On the other hand, there are some other mobile phones where the default Wi-Fi sleep policy is set as “never”, which means that Wi-Fi will be operating all the time since it is turned on by the user until it is turned off manually by the user.

References

[1] http://loclizard.com/products/loclizard/

[2] http://www.qubulus.com.

[3] A.K.M. Mahtab Hossain, Hien Nguyen Van, Yunye Jin, Wee-Seng Soh - *Indoor Localization Using Multiple Wireless Technologies.* IEEE Press, 2007.

[4] Eddie C.L. Chan, George Baciu, S.C. Mak - *Wireless Tracking Analysis in Location Fingerprinting.* IEEE Press, 2008.

[5] Rong-Hong Jan; Yung Rong Lee; , "An indoor geolocation system for wireless LANs," *Parallel Processing Workshops, 2003. Proceedings. 2003 International Conference on* , vol., no., pp. 29- 34, 6-9 Oct. 2003

[6] http://en.wikipedia.org/wiki/Trilateration

[7] Kundu Sudakshina – *Analog and Digital Communications*. Pearson, 2010.

[8] Jorg Eberspacher, Jörg Eberspächer, Christian Bettstetter, Hans-Joerg Vögel, Christian Hartmann, Hans-Joerg Vgel, Jö Rg Eberspä Cher - *GSM - Architecture, Protocols and Services.* Wiley, 2009.

[9] Andreas F. Molisch - *Wireless Communications.* Wiley/IEEE Press, 2005.

[10] Thomas Toftegaard Nielsen, Jeroen Wigard-*Performance enhancements in a frequency hopping GSM network.* Public Academic Publishers, 2000.

[11] 3GPP TS 25.304 V10.0.0 *User Equipment (UE) procedures in idle mode and procedures for cell reselection in connected mode.* 3GPP, 2011.

[12] 3GPP TS 43.022 V10.0.0 *Functions related to Mobile Station in idle mode and group receive mode.* 3GPP, 2011.

[13] Dahlman, E.; Beming, P.; Knutsson, J.; Ovesjo, F.; Persson, M.;

Roobol, C.; , "WCDMA-the radio interface for future mobile multimedia communications," *Vehicular Technology, IEEE Transactions on* , vol.47, no.4,pp.1105-1118,Nov1998

[14] Wikipedia IEEE IEEE 802.11g-2003
http://en.wikipedia.org/wiki/IEEE_IEEE 802.11g-2003.

[15] Branimir Boskovic, Milan Markovic *On Spread Spectrum Modulation Techniques Applied in IEEE IEEE 802.11 Wireless LAN Standard.* IEEE Press, 2000.

[16] Moustaffa A. Youssef, Raymond E. Miller – *Analyzing the Point Coordination Function of the IEEE IEEE 802.11 WLAN Protocol using a Systems of Communicating Machines Specifications.* University of Maryland, 2002.

[17] IEEE Standard IEEE 802.11g *Information technology — Telecommunications and information exchange between systems — Local and metropolitan area networks — Specific requirements — Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications.* IEEE Press 2003.

[18] Shao-Cheng Wang, Yi-Ming Chen, Tsern-Huei Lee, Ahmed Helmy – *Performance Evaluations for Hybrid IEEE IEEE 802.11b and IEEE 802.11g Wireless Networks.* Performance, Computing, and Communications Conference, IPCCC 2005.

[19] IEEE Standard IEEE 802.11 *Information technology- Telecommunications and information exchange between systems- Local and metropolitan area networks- Specific requirements- Part 11 : Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specification-* 1997.

[20] J. Seybold, *Introduction to RF Propagation*, Wiley Interscience , 2005.

[21] www.ettus.com

List of Figures

List of Tables

Appendix 1

A.1 Python code

It was modified and added some new code to the script developed by GNU radio. The script is generally found in gnuradio/gnuradio-examples/python/usrp/usrp_spectrum_sense.py.

It will be just shown the lines that were modified and the new methods and code developed.

```
# build graph
        if self.opc==1:
          self.u =
usrp.source_c(which=0,fusb_block_size=options.fusb_block_size,fusb_nbl
ocks=options.fusb_nblocks)
        if self.opc==2:
          self.u =
usrp.source_c(which=1,fusb_block_size=options.fusb_block_size,fusb_nbl
ocks=options.fusb_nblocks)
        if self.opc==3:
          self.u =
usrp.source_c(which=2,fusb_block_size=options.fusb_block_size,fusb_nbl
ocks=options.fusb_nblocks)

# Set the freq_step to 50% of the actual data throughput.
# This allows us to discard the bins on both ends of the spectrum.

self.freq_step = 0.5 * usrp_rate

# add the following new methods
def adjust_data(m_data,fft_size):   # adjust the fft from -Fs/2 to Fs/2
    var1=m_data[0:1]
    var2=m_data[1:fft_size/2]
```

```
   var3=m_data[fft_size/2:]
   var3.extend(var1)
   var3.extend(var2)
   return var3

def overlapping(f_cent_1,f_resol,n_freq,f_step,f_size,t_data):        #
Ovelapping to eliminate zeros in the fft
   f_cent1_max=f_cent_1+f_resol*(f_size/2-1)
   f_cent_2=f_cent_1+f_step
   f_cent2_min=f_cent_2-f_resol*(f_size/2)
   num_sample_up=int((f_cent1_max-f_cent2_min)/f_resol)
   #num_sample_low=(f_cent2_min-f_cent_1)/f_resol
   j=1
   var=t_data[0:f_size-1-num_sample_up]
   while j<n_freq:
      g1=array(t_data[(f_size*j-1)-num_sample_up:f_size*j])
      g2=array(t_data[f_size*j:f_size*j+num_sample_up+1])
                ii=0
                for x in g2:
                   if g1[ii]<=x:
          g1[ii]=x
          ii=ii+1
        else:
                      ii=ii+1
                #g_aver=list((g1+g2)/2)
                #g_aver=list(g1+g2)
                #var.extend(g_aver)
                var.extend(g1)
                var_h=t_data[f_size*j+num_sample_up+1:(f_size*(j+1)-1)-
num_sample_up]
      var.extend(var_h)
      if j==n_freq-1:
         aux=t_data[f_size*n_freq-1-num_sample_up:]
         var.extend(aux)
         j=j+1
      else:
         j=j+1
   return var

def write_to_file(data,y): # write to text file
```

```
    if y==1:
      f=open('/home/rick/collected_data/umts/umts_idle.dat','a')
    if y==2:
      f=open('/home/rick/collected_data/Wi-Fi/test/usrp2_45.dat','a')
    if y==3:
      f=open('/home/rick/collected_data/Wi-Fi/test/up_pos1_usrp3.dat','a')
    k=str(data)
    k=k.replace("[","  ")
    k=k.replace("]","  ")
    f.write(k)
    #f.write("\n")
    f.close()

# The part below you can just replace as it contains several changes and
new code

def main_loop(tb):

    while 1:

        # Get the n,ext message sent from the C++ code (blocking call).
        # It contains the center frequency and the mag squared of the fft

                i=1
                temp=[]
        temp1=[]
                temp2=[]
                usrp=8e6
        ##### set parameters
        freq_min=1.9654e9
        freq_max=1.9794e9
                fft_size=1024
        freq_step=4e6
        usrp2=False
        usrp3=False

        num_freq = math.ceil((freq_max - freq_min) / freq_step)
                num_freq = int(num_freq)
        #print num_freq
                freq_cent_min=freq_min + freq_step/2
```

```
                freq_resol=usrp/fft_size
        while i<=num_freq:
                    m = parse_msg(tb.msgq.delete_head())
          #print m.center_freq
          if usrp2:
                      m1 = parse_msg(tb1.msgq.delete_head())
                    if usrp3:
                      m2 = parse_msg(tb2.msgq.delete_head())
          #print m.data
          #print "------------------------------------------------------------"
          #print m1.data
                    m = list(m.data)
                    a_data=adjust_data(m,fft_size)
          temp.extend(a_data)
          if usrp2:
                      m1 = list(m1.data)
            a_data1=adjust_data(m1,fft_size)
            temp1.extend(a_data1)
          if usrp3:
            m2 = list(m2.data)
            a_data2=adjust_data(m2,fft_size)
                      temp2.extend(a_data2)
          if i==num_freq:

over_data=overlapping(freq_cent_min,freq_resol,num_freq,freq_step,fft_si
ze,temp)
            write_to_file(over_data,1)
            temp=[]
            if usrp2:

over_data1=overlapping(freq_cent_min,freq_resol,num_freq,freq_step,fft_s
ize,temp1)
              write_to_file(over_data1,2)
              temp1=[]
            if usrp3:

over_data2=overlapping(freq_cent_min,freq_resol,num_freq,freq_step,fft_s
ize,temp2)
              write_to_file(over_data1,3)
```

```
            temp2=[]
          i=1
        else:
          i=i+1



      #Print center freq so we know that something is happening...
      #print m.center_freq

      # FIXME do something useful with the data...

      # m.data are the mag_squared of the fft output (they are in the
      # standard order.  I.e., bin 0 == DC.)
      # You'll probably want to do the equivalent of "fftshift" on them
      # m.raw_data is a string that contains the binary floats.
      # You could write this as binary to a file.


if __name__ == '__main__':
    now = datetime.datetime.now()
    print str(now)
    tb = my_top_block(1)
    #tb1 = my_top_block(2)
    #tb2 = my_top_block(3)
    try:
        tb.start()          # start executing flow graph in another thread...
        #tb1.start()
        #tb2.start()
        main_loop(tb)
    except KeyboardInterrupt:
                now = datetime.datetime.now()
                print str(now)
        pass
```

Appendix 2

A.1 Parser code

The following code was developed in order to parse text files with data collected by the routers and get the information required by us.

```
import javax.swing.JFileChooser;
import javax.swing.JOptionPane;
import java.io.*;
import java.util.StringTokenizer;
import java.util.regex.Matcher;
import java.util.regex.Pattern;


class LogParser {

        public static void main (String[] args) throws IOException{
                int status;
                File file;
                String fileName;
                String str;

                //Open the FileChooser
                JFileChooser chooser = new
JFileChooser("/home/rick/collected_data/Wi-Fi_traffic/malformed");
                while(true){
                   status = chooser.showOpenDialog(null);
                   if(status ==
JFileChooser.APPROVE_OPTION){
                                        file =
chooser.getSelectedFile();
                                        break;
                                }else{

        JOptionPane.showMessageDialog(null, "Select a file");
```

```
                                        }
                                }

                                //
                                FileReader fileReader = new FileReader(file);
                                BufferedReader bufReader = new
BufferedReader(fileReader);

                                 // Get output fileName
                                fileName = JOptionPane.showInputDialog(null,
"Enter output file name for htc");

                                Matcher matcher;
                                Pattern pattern = Pattern.compile("Malformed",
Pattern.CASE_INSENSITIVE);

                                Matcher matcherTime;
                                Pattern patternTime =Pattern.compile("Power",
Pattern.CASE_INSENSITIVE);

                                Matcher matcherPower;
                                Pattern patternPower =Pattern.compile("-[0-
9][0-9]", Pattern.CASE_INSENSITIVE);

                                //
                                File outFile = new
File("/home/rick/collected_data/Wi-Fi_traffic/corregidos/"+fileName);
                                FileOutputStream outFileStream = new
FileOutputStream(outFile);
                                PrintWriter outStream = new
PrintWriter(outFileStream);

                                while(true){

                                        String line = bufReader.readLine();

                                        if(line == null) {
                                                outStream.close();
                                                break;
                                        }
```

```
                                        //
                                        matcher = pattern.matcher(line);
                                        matcherTime =
patternTime.matcher(line);
                                matcherPower = patternPower.matcher(line);

                                        boolean booleanMalformed =
matcher.find();
                                        boolean booleanTime =
matcherTime.find();
                                        boolean booleanPower =
matcherPower.find();

                                        if(false==booleanTime){

                if(false==booleanMalformed){

                System.out.println(line);

                StringTokenizer strTokenizer = new StringTokenizer(line);
                                                                        int j=2;

                while(strTokenizer.hasMoreTokens()){

                if(j==1){

                  outStream.print(strTokenizer.nextToken());

                  outStream.print(" ");

                  j++;
                                                                        }

if(j==2){

if(true==booleanPower){

                str = line.substring(matcherPower.start(),
matcherPower.start()+3);
```

```
outStream.print(str);
                    outStream.print("\n");
                    j++;
                                                                    }
                }
strTokenizer.nextToken();
                j++;
                                                                }
                                                        } else{
                System.out.println(booleanMalformed);
                                                        }
                            }
                        }
                }
}
```

Appendix 3

A.1 LYNSYS WRT54GL specifications

Model: WRT54GL.

Platform: Broadcom MIPS

CPU: Broadcom BCM5452 at 200 MHz (130nm construction)

Standards: IEEE 802 .3, IEEE 802 .3u, IEEE 802 .11g, IEEE 802 .11b.

Channels: 11 Channels (US, Canada), 13 Channels (Europe, Japan).

Ports: Internet: One 10/100 RJ-45 Port ; LAN: Four 10/100 RJ-45 Switched Ports, One Power Port.

Flash: 4 MB NAND, single chip.

System Memory: 16 MB 16-bit DDR SDRAM.

USB: None.

Wireless Radio: Broadcom BCM43xx IEEE 802.11b/g.

Antenna: Dual folding, removable, rotating antennas.

Button: Reset, SecureEasySetup.

Cabling Type: CAT5.

LEDs: Power, DMZ, WLAN, LAN (1-4), Internet, SecureEasySetup

RF Power Output: 18 dBm.

UPnP able/cert: Able.

Security Features: Stateful Packet Inspection (SPI). Firewall, Internet Policy.

Wireless Security: Wi-Fi Protected Access™2 (WPA2), WEP, Wireless MAC Filtering.

Dimensions: 186 x 48 x 154 mm.

Weight: 391 g.

Power: External, 12V DC, 0 .5A.

Certifications: FCC, IC-03, CE, Wi-Fi (802 .11b, 802 .11g), WPA2, WMM.

Operating Temperature: 0 to 40ºC.

Storage Temperature: -20 to 70ºC.

Operating Humidity: 10 to 85%, Noncondensing.

Storage Humidity: 5 to 90%, Noncondensing.

Information acquired at
http://downloads.linksysbycisco.com/downloads/userguide/1224639045490/WRT54GL-EU_11_UG_A-WEB.pdf.

Appendix 4

A.1 Getting started with dd-wrt

Requirements

- A computer (Windows, Linux, Mac, whatever).
- A broadband internet connection (DSL, Cable, or similar).
- A Linksys WRT54G/GL/GS router or other supported router.
- The DD-WRT firmware image from The DD-WRT Project.

Information acquired at http://www.dd-wrt.com/wiki/index.php/What_is_DD-WRT%3F.

Installation process from a stock Linksys firmware

1. READ the peacock announcement linked above.
2. Do a hard reset.

You can HARD RESET by holding down the reset button on the back of the router for 30 seconds, then pulling the power cord for 30 seconds while STILL holding the reset button, and then plugging in the power cord for a final 30 seconds while STILL holding the reset button. You will hold the reset button in for 90 seconds without releasing it. Then release the reset button and wait for the router to finish doing whatever it's going to do. Usually the WLAN light will come on close to last in the boot sequence. Sometimes, however, the POWER light will keep flashing for a good while. Either way, once you're sure the router has done its thing, power cycle the router, by unplugging and re-plugging the power connector in the back of the router. There's no need to wait between unplugging and replugging.

3. Download the NEWD_MINI build generic 12548 Eko Build from here: ftp://dd-wrt.com/others/eko/V24_TNG/svn12548/dd-wrt.v24-12548_NEWD_mini.bin

4. You should check the MD5 HASH of the firmware after downloading it if it is available for your build. See Hashes & Checksums. Turn off and disable your firewall, turn off and disable your antivirus, and sign into your linksys router with your web browser.

5. Use the firmware upgrade web interface to update your router with dd-wrt. **DO NOT close your browser; DO NOT interrupt the process, be EXTREMELY PATIENT, even after the firmware is already supposedly upgraded.** Wait around for a while, and make sure it settles down and is definitely finished doing whatever it's going to do. The router needs time to rebuild the NVRAM after it has been flashed, and if you interrupt this you will regret it!

6. When three or so minutes pass and the WLAN light comes back on, try and open the dd-wrt page at 192.168.1.1. If it opens, the process is complete. You should be asked to reset the password. Type username and passphrase in carefully.

7. Decide if you would like to keep the MINI version, or change to MICRO, STANDARD, VOIP, or VPN versions. **DO NOT try to load a MEGA build on this router (see above)**. If you are keeping MINI, you are done, otherwise, continue

Read about the different versions' features here: What is DD-WRT?#File Versions. If you won't be needing the features in the larger versions such as standard, you may be able to increase the responsiveness of your router by getting the smallest version that includes the features you need. Also, you can always update to a larger version later if down the line you need the extra features.

8. Power cycle the router.

9. Do a hard reset.

10. Install the version of dd-wrt you want.

11. Wait again for the process to complete and the lights to return to normal, and the webgui to be accessible at 192.168.1.1 showing the updated firmware. Give it at least 3 to five minutes.

12. Power cycle the router.

13. Do a hard reset

Other install notes

If you are upgrading from the web interface, you should use the GENERIC versions. For updating by webgui, EKO build 12548 Newd_Mini.bin is the recommended build for this router. It works well. You can also upgrade to 12548 Newd_Std.bin AFTER you have put on the mini version. Here is a link to the mini version download:
ftp://dd-wrt.com/others/eko/V24_TNG/svn12548/dd-wrt.v24-12548_NEWD_mini.bin.

Information acquired at http://www.dd-wrt.com/wiki/index.php/Linksys_WRT54GL

For further information visit the official DD-WRT website http://www.dd-wrt.com/site/index.

Appendix 5

A.1 Code to install tcpdump on DD-WRT routers

```
#!/usr/local/bin/expect

spawn telnet 192.168.1.1 % The IP address here is the router IP ad dress.
In our case we have 192.168.1.1, 192.168.1.2 and 192.168.1.3
expect "login:"
send "root\n"
expect "Password:"
send "admin\n"
expect "$>"
send "wl monitor 1\r"
send "mount --bind /tmp/smbshare /jffs\r"
send "nvram set sys_enable_jffs2=1 \r"
send "mkdir -p /tmp/smbshare/tmp/ipkg\r"
send "cd /tmp/smbshare/tmp/ipkg\r"
send "wget http://downloads.openwrt.org/whiterussian/packages/libpcap_0.9.4-
1_mipsel.ipk\r"
send "ipkg -d smbfs install libpcap_0.9.4-1_mipsel.ipk\r"
send "wget http://downloads.openwrt.org/whiterussian/packages/tcpdump_3.9.4-
1_mipsel.ipk\r"
send "ipkg -d smbfs install tcpdump_3.9.4-1_mipsel.ipk\r"
```

Appendix 6

A.1 Getting started with the USRP

This is a short guide of how to install GNU radio.

1) Build the USRP with the daughter boards you require for your application
2) Install Ubuntu on your computer and download updates
3) Open the terminal and copy the following (for ubuntu 10.04):

```
sudo apt-get -y install libfontconfig1-dev libxrender-dev libpulse-dev swig g++ automake autoconf libtool python-dev libfftw3-dev \

libcppunit-dev libboost-all-dev libusb-dev fort77 sdcc sdcc-libraries \

libsdl1.2-dev python-wxgtk2.8 git-core guile-1.8-dev \

libqt4-dev python-numpy ccache python-opengl libgsl0-dev \

python-cheetah python-lxml doxygen qt4-dev-tools \

libqwt5-qt4-dev libqwtplot3d-qt4-dev pyqt4-dev-tools python-qwt5-qt4
```

4) Install GNU radio from git

```
git clone http://gnuradio.org/git/gnuradio.git

cd gnuradio

./bootstrap

./configure

make
```

5) if you have USRP1

```
sudo addgroup usrp

sudo usermod -G usrp -a <YOUR_USERNAME>

echo 'ACTION=="add", BUS=="usb", SYSFS{idVendor}=="fffe",
SYSFS{idProduct}=="0002", GROUP:="usrp", MODE:="0660"' >
tmpfile

sudo chown root.root tmpfile

sudo mv tmpfile /etc/udev/rules.d/10-usrp.rules
```

6) you can check if the USRP is connected with the following command

```
ls -lR /dev/bus/usb | grep usrp
```

The result should appear as:

```
crw-rw---- 1 root usrp 189, 514 Mar 24 09:46 003
```

7) Once you have verified that the USRP is connected to the computer, Now, it is time to check if the USRP works, Run the following scripts:
Python interface:

```
cd gnuradio-examples/python/usrp

./usrp_benchmark_usb.py
```

C++ interface:

```
cd usrp/host/apps

./test_usrp_standard_tx
```

```
./test_usrp_standard_rx
```

If both have work, you are now ready to use GNU radio and USRP.

THE USRP

The Universal Software Radio Peripheral (USRP) connects to the computer through USB. It is used for making software radios. Regarding your application, you just need the right daughter board that should be plugged into the either RX or TX slots on the USRP.

The USRP runs at 64 MHz, however, the computer cannot receive this information fast enough. Using decimation is a way in which it can be reduced the number of samples retrieve from the USRP. According to [3] it can retrieve 8 MHz from the USRP due that the USB limitations.

There are new versions of the USRP, which are faster and can retrieve 16 MHz from the USRP.

If you want to sense the spectrum you can run the following command: usrp_spectrum_sense.py (generally found in gnuradio/gnuradio-examples/python/usrp/usrp_spectrum_sense.py). The output of running the command is explained below:

Assume you have N=1024 (I and Q) samples gathered using a tuner center at 896.7 MHz. The sampling frequency is 8 MHz. Doing 1024 points complex FTT means [21]:

- Frequency resolution is: 8MHz /1024 = 7.8125 KHz.
- The output of the FFT x[0] represents the spectrum at 896.7 MHz.
- The output of the FFT x[1] to x[511] represents the frequencies from 896.7078125 to 900.6921875 MHz.

The output of the FFT x[512] to x[1023] represents the frequencies from 892.7078125 MHz to 896.6921875 MHz