Escola d'Enginyeria de Telecomunicació i Aeroespacial de Castelldefels
UNIVERSITAT POLITÈCNICA DE CATALUNYA

MASTER THESIS

TITLE: Frequency management in a campus-wide Wi-Fi deployment

MASTER DEGREE: Master in Science in Telecommunication Engineering & Management

AUTHOR: Ester Mengual Pérez

DIRECTOR: Eduard Garcia Villegas

DATE: February 18, 2013

Título: Frequency management in a campus-wide Wi-Fi deployment

Autor: Ester Mengual Pérez

Director: Eduard Garcia Villegas

Fecha: 18 de febrero de 2013

Resumen

Desde hace unos años, Internet se está extendiendo cada vez más. Y no sólo en los hogares, sino también en aparatos de uso personal que están convirtiendo nuestro estilo de vida en un estilo "móvil".

Al principio de su aparición, la tecnología cableada era la más recurrida. Sin embargo, en este mundo cada vez más móvil, nos encontramos con que las redes cableadas no pueden satisfacer los nuevos retos. Es por esto que las redes inalámbricas están invadiendo el espacio de las tecnologías de acceso tradicionales. De hecho, la banda ancha móvil se está acoplando a un ritmo vertiginoso en muchos aspectos de nuestra vida; y esto parece ser sólo el comienzo, como demuestran los varios millones de personas que utilizan los aparatos tecnológicos más modernos.

Una de las tecnologías clave que permiten acceso móvil a Internet es IEEE 802.11, más conocida como Wi-Fi, término comercial que asegura interoperabilidad y compatibilidad entre productos. Parte de su popularidad se debe al uso del rango de frecuencias libres de licencia. Tradicionalmente, sólo tres de los catorce canales que establece la banda de 2,4GHz son asignados. Esto ha sido siempre así para evitar interferencias, ya que estos canales (1, 6 y 11) pertenecen a frecuencias que no se solapan. Sin embargo, la reciente explosión en el número de dispositivos inalámbricos así como la cantidad de protocolos que trabajan en el mismo rango frecuencial, provocan una saturación de este espectro limitado, haciendo así que se asignen los mismos canales en celdas cercanas. Sin duda, esto se traduce en interferencias, y en general, hace que el comportamiento de la red no sea el óptimo.

En este contexto, en el que el número de usuarios es elevado pero los recursos radio escasos, la asignación eficiente de canales se vuelve crucial para tener buen despliegue y funcionamiento de las redes WLAN basadas en IEEE802.11.

Y éste es de hecho el propósito principal de esta *Master Thesis*, que presenta un sistema capaz de establecer comunicación con una controladora y decidir una nueva distribución de canales según un conocido algoritmo matemático. El sistema que se presenta ha sido probado en el campus CBL–UPC Barcelona Tech, obteniendo resultados y conclusiones satisfactorios. Pero antes de entrar en los detalles del proceso, este trabajo también analiza los estándares IEEE 802.11 así como las técnicas actuales utilizadas en la gestión de recursos, de manera que el lector entenderá fácilmente el contexto en el que este trabajo se realiza y cómo el sistema presentado es capaz de mejorar una tecnología ya asentada.

Title: Frequency management in a campus-wide Wi-Fi deployment

Author: Ester Mengual Pérez

Director: Eduard Garcia Villegas

Date: February 18, 2013

Overview

Over the past years, Internet is spreading more and more. And not only in home devices, but also in other personal devices thus producing a collective mobile lifestyle.

At the beginning of its rise, cable solution was the most applied technology. However, in a world becoming a “mobile world”, wired networks can’t fulfil all new challenges. Thus, Wireless networks are increasingly encroaching on the niche of traditional access technologies. In fact, mobile broadband is exponentially being integrated into every aspect of life; and that seems to be only the beginning of what is to come, as the several millions of people using the latest trend gadgets ratify.

One of the key technologies that enable mobile Internet access is IEEE 802.11, commonly known by its trademark, Wi-Fi, which assures interoperability and backward compatibility between products. Part of the popularity achieved by Wi-Fi is due to the use of unlicensed (i.e. free) spectrum. Traditionally, only three of the fourteen frequency channels established in the 2.4GHz are used. This is so in order to avoid interference, as these channels (1, 6 and 11) belong to non-overlapping frequencies. However, the recent explosive growth in the number of wireless devices together with the multitude of wireless protocols operating in the unlicensed radio spectrum bands and sharing the same spectrum, lead to a saturation of the limited available spectrum, thus causing the frequency channels assigned to be repeated in close cells. Definitely, it results in interference and performance degradation, and broadly, a non-optimal performance of the network.

Within this context, in which the number of users is very large but the radio resources are scarce, efficient channel allocation becomes crucial for the successful deployment and operation of IEEE802.11-based WLANs.

And this is indeed the main purpose of this Master Thesis, in which a system able to establish communication with a controller and set a new channel distribution according to a renowned mathematical algorithm is exposed. The presented system has been tested in CBL – UPC Barcelona Tech campus, giving satisfactory outcomes and conclusions as a result. But before going deep into the process details, this work also reviews WLAN 802.11 standards and radio resource management techniques used nowadays, so the reader will easily understand the context which encompasses this work and how the system presented is able to improve an already settled technology.

First of all, I would like to express my gratitude to my advisor, Eduard Garcia, who has always been willing to solve any doubt I have had and has reviewed every line of this work.
Special thanks to the management of CBL, who believed in this project; and to UPCnet. Thanks for making this possible.

Finally, thanks to those who have been next to me along this stage, turning little moments into big ones. To the friends I have made and the ones who have been since before. And of course to my family, for supporting my efforts and celebrating my successes.
Everyone, THANKS!

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1. INTRODUCTION

Nobody foresaw that that message sent at the end of 1969 might mean for humanity as much as it meant. That day a group of researchers at the University of California had a computer connected to an Interface Message Processor (IMP), which made contact to a second computer located several miles away from there. And they managed to send a message from one to the other.

That moment is considered to be the beginning of the Internet. Since then, the number of devices has grown from a low number of scientists' computers up to near five thousand million devices connected to the Network of Networks. Not just computers take part of it, but nowadays we can find a huge variety of devices with Internet access: laptops, tablets, smartphones or ebooks are just some examples.

But the reach of such an amazing success has been largely due to the use of wireless technologies. IEEE 802.11 WLANs are increasingly becoming a popular solution to provide access to the Internet, especially in places such as airports, coffee shops and shopping malls, since it presents advantages that make it particularly attractive. Among others, its ease of installation and extension avoiding complex cable installation stand out, as they lead the users to have more freedom of movement while they keep connected, promoting a new step towards this collective “mobile” lifestyle.

For sure one of the main reasons behind the success of these technologies resides in the fact that they all work in the Industrial Scientific Medical (ISM) frequency bands, which belong to a small unlicensed part of the spectrum. In other words, they can be freely used by any device. However, this characteristic becomes at the same time its major drawback, since this small band of frequencies is shared by a growing number of users. Moreover, these bands are also shared by different wireless protocols that may operate at the same time, which hampers any kind of synchronization or resource management and leads to interference and performance degradation for all the protocols working in this unlicensed band.

In such a world we live in, where people's demands for faster communications is accelerating, both interference and performance degradation must be fought so that current WLAN technologies can deliver its maximum performance. Within this scenario, the efficient use of radio spectrum resources acquires importance. The aim of radio resource management techniques is to palliate these negative effects through strategies and algorithms for controlling relevant parameters (channel allocation, load balancing, data rate…).

This work is intended to enter the world of WLAN radio resource management and explore different channel management techniques. Among several studies in this field, it is not common to address the issue in a real environment. Therefore, this work presents a system designed to work in a campus size environment using a modified version of DSATUR graph coloring algorithm. As in any radio resource management technique, the system presented tries to improve network performance and capacity. To ascertain to what extent it provides enhancements, the system is tested under real traffic conditions to provide a collection of SNMP statistics that are analyzed to get quantifiable measures of the results.

1.1. Project description

The idea of this Master Thesis is born from [1], where a new frequency management scheme for IEEE 802.11 WLANs is introduced. Summing it up, the proposed model basically adapts a weighted Degree of Saturation (DSATUR) algorithm to better fit the characteristics of a WLAN. Although successfully evaluated, the improvements it provides are quantified by means of simulations and a small testbed. Thereupon, a real environment test was still pending in order to prove that its performance continued being the expected in a “non so controlled” environment.

That being said, Campus Baix Llobregat (CBL) [2], which is part of the Universitat Politècnica de Catalunya (UPC) Barcelona Tech [3], was a perfect model to carry out this study, both for its campus-size dimensions, its high number of users and also for its influence in our daily work, as it is where this engineering school is located.
After an agreement between EnTel (Enginyeria Telemàtica) department [4] and UPCnet [5] (campus Wi-Fi provider), we have adjusted our approach to be suitable for such a campus centralized wireless system.

Six chapters compose this work; the first of which corresponds to this introductory chapter. In it a little tour through wireless Internet evolution is presented so that the reader gets introduced into the core of the matter.

The rest of the contents are organized as follows. The second chapter deals with the main IEEE 802.11 standards and the basics of WLAN operation. After the basis of Wi-Fi technology, this chapter focuses on enterprise WLAN management, and takes a glance at the method used to extract and set key configuration parameters that is detailed in future stages. Additionally, this chapter includes a comprehensive state-of-the-art review.

Chapter 3 contains the description of the environment where this study takes place. Besides describing the location of the campus from a general perspective, it also deals with detailed hardware and software features. On the one hand, it addresses the description of those devices composing the studied network. On the other hand, it also presents specifications of useful software for the development of this work.

Next, chapter 4 is where implementation details can be found. Firstly, the approach used to perform an efficient radio resource management is deeply explained. The description of the system designed also occupies an important portion of this chapter, which is simply explained through an example scenario.

In the last stage of the process the system has been run with success. An analysis of the information extracted during the time it has been working is shown in chapter 5. There, several aspects related to network performance are compared using information from before and after applying our approach.

This work is closed with chapter 6, which explains conclusions and future lines of work drawn during the design and implementation of the system. This last chapter provides the final considerations and check that initial goals have been successfully fulfilled. Also, some possible lines of work are proposed so that the research done in this field continues.

CHAPTER 2. FREQUENCY MANAGEMENT IN 802.11 WLANs

This second chapter begins with a look at IEEE 802.11 standard and its outstanding amendments. It also analyzes the basics of Wi-Fi technology, with special attention to the management of enterprise networks, and how important is to do it suitably. This chapter comes to an end with a review of the different techniques that have been proposed with the aim of improving wireless networks performance through channel management.

2.1. IEEE 802.11 Wireless LANs

Commonly known as Wi-Fi technology, IEEE 802.11 standard started being specified in the nineties through IEEE. Its purpose was to define how computers in a LAN should safely interact with each other wirelessly. As a result, a non-profit organization in charge of certifying Wi-Fi products according to interoperability, backward compatibility and innovation emerged. Its name is Wi-Fi Alliance [6].

Fig. 2.1. Wi-Fi icon

This section aims to give an overview of IEEE 802.11 standard. Firstly, we present a description for a selected set of amendments, those distinguished because of either its historical importance or their connection with frequency management. Then, the basics of IEEE 802.11 WLAN operation are presented.

2.1.1. Technologies

The initial IEEE 802.11 standard was released in June, 1997.That standard specified data rates of 1 and 2 Mbps operating in the 2.4 GHz ISM band obtained through the use of Frequency Hopping Spread Spectrum (FHSS) or Direct Sequence Spread Spectrum (DSSS). Afterwards, several working groups (WG) were created, giving a large list of amendments as a result. Some examples are 802.11a, 802.11b and 802.11b/g; whose letters in addition to the generic name represent the addition of different features. The whole set of amendments is often referred to as 802.11 alphabet soup.

Fig. 2.2. IEEE icon

Despite the large number of amendments, they all share a constant target, achieving a good balance between three important factors: network performance, operation frequency and bandwidth [7]. Of course, each evolution tries also to improve issues related to security enhancements.
Among the wide assortment, it should be noted the ones explained by chronological order below.

***802.11a**: OFDM in the 5 GHz Band.* Ratified in 1999, this amendment belongs nowadays to the standard. It describes an OFDM-based physical layer standard for wireless stations operating in the 5GHz band. It supports data rates ranging from 6 to 54Mbps. The fact of working in a high frequency implies that signals have more problems to penetrate obstructions, e.g. walls; in consequence the signal range is shortened in comparison to the original 2.4GHz technology.
***802.11b**: 802.11 High Rate*. It was proposed in 1999 as an enhancement of the original standard in order to allow 5.5 and 11 Mbps operating in the 2.4 GHz band. The use of a lower frequency than 802.11a reduces production costs, but on the other hand, it shares the same spectrum as many devices working also in the same unregulated range. Most products comply with it, as it is the basis for Wi-Fi certification granted by Wi-Fi organization [6].
***802.11g**: Up to 54 Mbps in the 2.4GHz*. This amendment is compatible with 802.11b standard, and as with 802.11b products, 802.11g products are also considered Wi-Fi certified. Actually, 802.11g has been progressively replacing 802.11b. 802.11g specification establishes throughput up to 54 Mbps with the same OFDM modulation scheme defined in 802.11a using the same 2.4 GHz band as 802.11b.
***802.11h**: Spectrum and transmit power management.* When it came out in 2003, IEEE 802.11h was intended to provide Dynamic Frequency Selection (DFS) and Transmit Power Control (TPC) so it could solve problems like interference with satellites and radar present in the 5GHz band. DFS ensures radar detection and avoidance by an AP. This way, energy is spread across the band to reduce interference to satellites. TPC is used for managing client transmit power, so it complies with the transmission power limitations applicable in a given region. In other words, this amendment provides useful means for the frequency management, reason why it is relevant in the context of this thesis.
***802.11k**: Radio resource management.* IEEE 802.11k amendment was released in 2009 with the aim of providing client feedback to wireless infrastructure. It defines a set of measurement requests and reports (roaming decisions, hidden nodes, client statistics, Transmit Power Control) that can be exchanged between any stations. Because it is designed to be implemented in software, the standard needs not just infrastructure but also clients to support it in order to be effective.
***802.11n**: Higher throughput through MIMO.* This newfangled amendment (2009) defines PHY and MAC modifications so that much higher throughputs (up to 600 Mbps) can be reached. This is obtained by exploiting multiple-input-multiple-output (MIMO) techniques, allowing channel bonding and frame aggregation. MIMO allows multiple data streams to be sent simultaneously along the same channel taking advantage of the multiplicity of data streams to improve data rate as well as at greater distances. Channel bonding bonds two adjacent 20MHz channels together, so it provides the option of operating over a 40 MHz channel, which doubles the PHY layer data rate.

New features already mentioned allow significant increases in data rate. Even potential characteristics, they can only been achieved through intelligent and adaptive channel management strategies [8].

***802.11ac**: Beyond the barrier of 1 Gbps.* This new amendment is currently under development and although some specifications are known, its approval is expected by the end of 2013. The amendment will enhance the 802.11n MAC and PHY to support very high throughput (500-1000 Mbps) in the 5 GHz band. This will be accomplished by the combination of Very High Throughput (VHT) with gigabit speeds, building on 802.11n MIMO technology.
With regards to frequency management, the most notable feature is that channels double their maximum allowed bandwidth in relation to 802.11n (from 40 to 80 MHz). This enhancement allows an increase of performance in a cost-efficient way.
Of course, this amendment will provide backward compatibility with previous ones, an essential feature taking into account the large number of devices working in the same frequency range.

Hence, IEEE 802.11 represents nowadays a family of specifications, all sharing the air interface but with different characteristics between them. The most recent amendment of the standard already published at the time of writing this work is 802.11-2012 [9] which incorporates those 802.11 amendments ratified after 2008, thus integrating all current Wi-Fi technologies into a single one.

Having reached this point, we can get the feeling that the huge quantity of technologies with different needs of radio resources coexisting together call for a good management of the frequency spectrum, becoming an increasingly important issue in network management.

2.1.2. Elements of a Wi-Fi-based WLAN

Taking a glance to any WLAN, there are two major physical components: the wireless client (also called station) and the Access Points which communicate through the wireless medium.

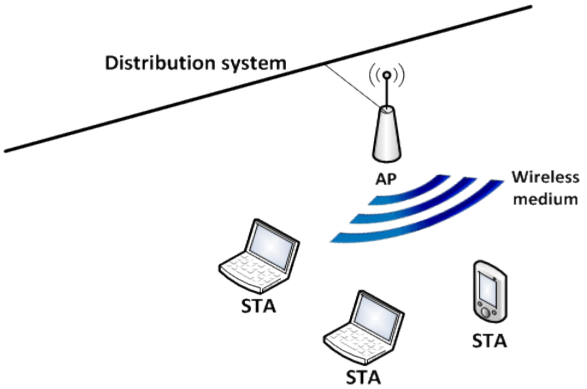


Fig. 2.3. WLAN elements

A *station* (STA) is the end element in a WLAN, insomuch networks are built to transfer data between them. It can be any device containing a 802.11 Medium Access Control (MAC) and Physical layer (PHY) interface (IEEE 802.11-conformant card or chip), as well as a client adapter so that communication over a radio link can be carried out.

Data from one STA to another one are transported through the wireless medium. In infrastructure-based WLANs, the other side-element in the radio link is always an *Access Point* (AP). It includes the same functionalities of a STA as well as the capability to reach the wired LAN, since they perform the wireless-to-wired bridging functions. Therefore, it can provide access to distribution services for associated clients.

In addition, two wireless devices that want to communicate directly without the need of an intermediary AP can operate in ad-hoc mode, which allows nearby STAs to discover and establish communication.

A group of STAs under the control of a given AP comprise the basic building block of an IEEE 802.11 WLAN, which is called Basic Service Set (BSS). This area usually depends on the coverage area of the AP, and therefore, it varies with the propagation characteristics of the wireless medium. The set of STAs share a common identifier, the Basic Service Set Identifier (BSSID). In a broader sense, networks are based on a cellular architecture where the system is subdivided into cells, each controlled by an AP in charge of a BSS.

When multiple BSSs are interconnected sharing a common distribution system they form an Extended Service Set (ESS). Each AP forming part of the ESS broadcast the same SSID, called Extended SSID (ESSID).

Each AP uses a different frequency which differs depending on the 802.11 technology used. In order to avoid interference with neighbouring cells, an intelligent frequency management mechanism is required. When the APs belonging to the same network are managed in a centralized way, there is the need of having a Central Controller (CC) or just called *Controller***.** This is precisely the situation on the scenario under study as detailed in section 2.3.

2.1.3. Operation

When a STA wants to join an AP, and therefore its cell, it needs to scan the air medium in order to get information about synchronization. This scanning can be a passive scanning (the STA waits for the reception of a *Beacon* Frame with synchronization information sent periodically by the AP) or an active scanning (the STA tries to find an AP by transmitting *Probe Request* Frames and waiting for a *Probe Response* from it). Active scanning is a faster process, unless the network has few channels and short beacon intervals.

Once a STA has found an AP, there are still two pending processes: authentication and association. In the *Authentication Process* there is an interchange of information between both user and AP, often a shared-key authentication concerning the BSSID. In case of being an open-system, this process is much simpler, as any station request is allowed. During the *Association Process* both the authenticated STA and the AP interchange technical information. Association allows the distribution system to track the location of any STA, so data frames will be delivered to the corresponding AP.

When the process is completed, then both the AP and the STA are prepared to transmit and receive data.
In the course of the connection between a STA and a given AP, often happens that the STA moves from one cell to another while a data session is active and hence any disruption should be avoided. Either because of mobility or because of interference, the standard allows roaming between APs either in the same or in different channels. However, it does not specify the exact procedure for this purpose [10] but generally it occurs when the signal strength received at the STA end goes under a given threshold.

2.1.4. Access method and collision avoidance

The mechanism used to access the medium works in the following way: when a station wants to transmit, it senses the medium; if the medium is free then the station is allowed to transmit; however, if another station is transmitting, then the station will postpone its transmission. The time each station waits until it tries to transmit again is set by a random backoff procedure. This way, those collisions due to simultaneous transmissions of multiple stations waiting for the medium to become idle are avoided.
When a collision occurs, the collided packet must be retransmitted. These situations are identified by the MAC layer thanks to its Stop&Wait nature. After an unicast transmission, the transmitter expects the reception of a positive Acknowledgement (ACK) frame (it does not happens for multicast) frames. If no ACK is received, a retransmission will be scheduled by the sender after doubling the contention window, i.e. after doubling the range of values over which the backoff timer is randomly chosen.

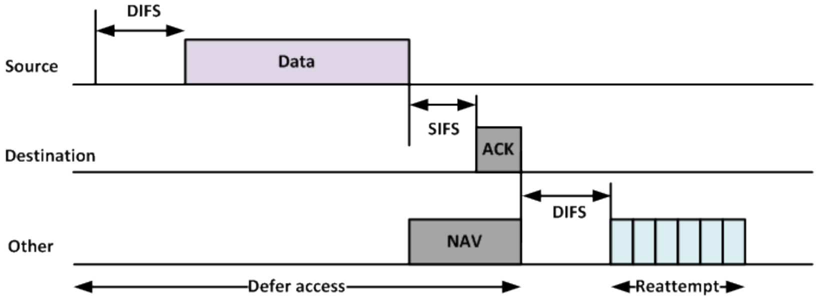


Fig. 2.4. IEEE 802.11 basic operation

In order to reduce the number of collisions, a three-way access is also defined: when a station wants to transmit a packet, it first sends a short frame RTS (Request to Send) and waits for a CTS (Clear to Send) response from the destination station, meaning that the medium is free, so that the wireless medium will become reserved until the end of the transaction.

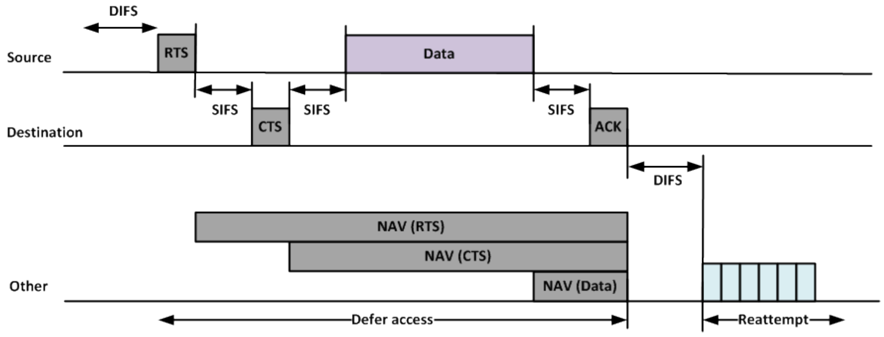


Fig. 2.5. IEEE 802.11 access using RTS/CTS

This mechanism to control access to the medium is known as Carrier Sense Multiple Access with Collision Avoidance (usually written CSMA/CA).

Due to its mode of operation, effectiveness of CSMA/CA mechanism gets reduced when two neighbouring cells share the same frequency channel. Indeed, two co-channel cells do not suffer from an increased SINR but instead, they have to coexist together sharing the same medium.

Hence, channel management in CSMA/CA-based WLANs becomes a particularly interesting matter to consider regardless of the scenario. It is an essential issue to study in order to improve the general network performance.

2.2. Enterprise WLAN management

With the rapid deployment of Wi-Fi hotspots in recent years, wireless LANs (WLANs) have become the major technology to provide wireless Internet access in places such as office buildings, shopping malls, airports and campus-size areas. The term *enterprise* will be used along this work to cover all of them.

However, enterprise managers need wireless networks to include performance criteria similar to wired enterprise networks. In fact, an ideal WLAN would be that providing similar reliability and availability to what wireless clients are already used to.

Anyway, management of WLANs becomes an important issue for any network manager. With the aim of easing the management of large WLAN networks, different software solutions allowing monitoring and controlling network devices can be found on the market. They are known as Network Management Systems (NMS).

Despite the variety of available NMS, they all share similar characteristics and features. Any NMS should be able to support firmware/software management, configuration management and performance reporting. Also *syslog* messages and *traps* are useful to obtain notifications about the system.

The NMS is basically formed by a manager station and an agent, which are explained in detail below:

A *manager station* is a user interface which makes possible the monitoring and controlling of the network to an operator. Apart from monitoring and controlling, a manager also organizes, summarizes and simplifies the information taken

from agents. At least one of the nodes involved in a network must have this role.
A management *agent* makes reference to the software running on the managed nodes. Its purpose is to obey orders asked by the manager station (which is mainly related with data collection or configuration setting), and compile statistics on that information locally stored on the Management Information Base (MIB). Besides, it informs the control centre when a significant change in the node occurs.

The communication established between these two elements is generally accomplished through the standardized SNMP (Simple Network Management Protocol), a network management protocol at the application layer. This protocol eases the exchange of information related to network monitoring. In fact, it is nowadays the standard for management and administration of IP tools.

2.2.1. SNMP

As explained before, different application software sharing similar characteristics and features exist. Most of those tools use SNMP communications in order to be able to establish communication with various agents and devices (routers, switches, access points…) from different manufacturers and vendors supporting the standard SNMP.
SNMP is a widely used protocol defined by Internet Engineering Task Force (IETF) for monitoring and management of IP networks. The most recent version available is SNMPv3, although SNMPv1 and SNMPv2 are still in use.

In an SNMP-enabled network, entities communicate through command and report messages. These messages are UDP (User Datagram Protocol) datagram containing a version identifier, the SNMP community name and a PDU (Protocol Data Unit). Among the set of command operations allowed, the most relevant are: *get*, *walk* and *set*.
The database used to check if all devices on the network are operating properly is called Management Information Base (MIB). Each one of the objects comprising the MIB have a unique object identifier (OID), and can be either dynamic (values are modified by the manager station) or static (include data about statistics and configuration parameters). There is a wide variety of MIBs, each one built for a given purpose, either for performance management or related to a specific technology. They all are classified in a hierarchical way, in which different levels are assigned to different organizations. Thus, top level OIDs belong to standard organizations such as Institute of Electrical and Electronics Engineers (IEEE) or International Organization for Standardization (ISO).
Within the context of WLANs, there is the 802.11 MIB defined by the IEEE 802.11 working group. The Abstract Syntax Notation (ASN.1) that describes this branch of the MIB tree is .1.2.840.100036, which refers to *body.us.ieee802dot11.* It is composed of five branches (smt, mac ,res, phy, conformance) as seen in the figure below.

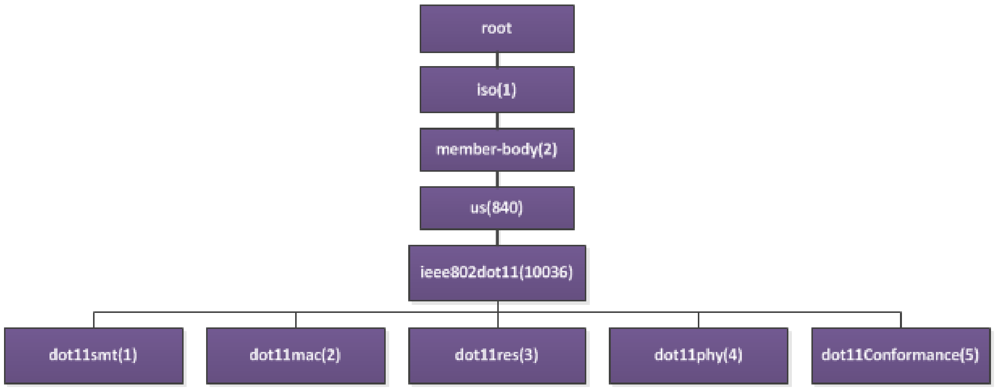


Fig. 2.6. SMI tree for ieee802dot11 group

Among those, MAC layer (dot11mac) parameters are the most interesting ones for the purpose this thesis covers. Examples of objects contained here are: *MACAddress*, *TransmittedFragmentCount*, *FailedCount* and *RTSSuccessCount* which report the status and performance of MAC parameters and allow its configuration.

Besides the standard MIBs, enterprises usually define “private” branches so they can adapt their needs and differentiate their products. This is the case of Airespace, Inc. owner of Airespace-Wireless-MIB [11], which provides configuration and status information for 802.11 Access Points and intends to be implemented on devices operating as Controllers.

2.2.2. Centralized WLANs

With the spread of networks and the increasing number of functionalities they provide, many enterprises choose a centralized LAN structure for their wireless network deployment. Thus, a controller implements most of the management and configuration logic, while APs are just responsible for communicating with end users. APs designed to be managed by a Wireless LAN Controller (WLC) are known as Lightweight AP (LAP).

Having centralized LANs allows administrators to better analyze the network behaviour. Moreover, having a single device for controlling so many functions reduces the amount of time spent on configuring, monitoring and troubleshooting the network. Furthermore, controllers act as manager stations, providing SNMP statistics about associated APs, and even graphical information of total traffic information, associated users, interference, etc.

After considering other protocols, Cisco decided to develop a protocol to transport data and facilitate communication between LAPs and WLCs. It was called LWAPP, which evolved towards CAPWAP. Both protocols are explained in the subsections below.

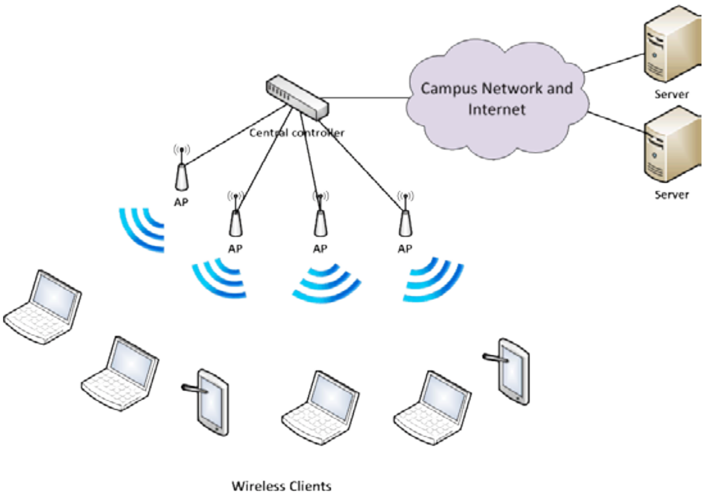


Fig. 2.7. Centralized WLAN architecture

2.2.2.1. LWAPP

The term LWAPP stands for Light Weight Access Point Protocol, a generic protocol (IETF RFC 5412) defined with the aim of facilitating centralized management and automated configuration of LAPs.
This protocol, originally proposed by Airespace, defines how access points communicate with access controllers, more specifically LWAPP defines the process of authentication between and AP and a controller, how controllers distribute firmware and configuration among LAPs, and also the transport header for LWAPP traffic. Communication between them can be either by means of Layer 2 or Layer 3.
The main target of LWAPP was to solve the specific problem of associating APs to controllers and managing firmware and configuration updates. However, LWAPP introduced many security issues on its design what lead to the development of an update, CAPWAP.
LWAPP has been implemented in Cisco WLCs until release 5.2. After that, CAPWAP has been the standard incorporated into them.

2.2.2.2. CAPWAP

In order to communicate controllers and lightweight APs, the management system may use the IETF Control And Provisioning of Wireless Access Points (CAPWAP).
CAPWAP [12], which is based on LWAPP, is a standard that enables a controller to communicate with each AP it manages. Not just communication, but a controller manages configuration, firmware, control transactions and data transactions from any of the APs joining it. Communication is done through UDP ports 5246 and 5247.
In the discovering process, a lightweight AP sends a CAPWAP join request to the controller, which is answered with a CAPWAP join response allowing the AP to join the controller. From this moment, the controller takes ownership of the

AP or can also redirect control to another controller. This process is the same as in LWAPP, one of the similarities CAPWAP has with its predecessor. CAPWAP also inherits the differentiation between data traffic and control traffic and the way it manages APs' configuration. However, CAPWAP doesn't support LWAPP's Layer 2 deployments. In addition, CAPWAP includes Datagram Transport Layer Security (DTLS) defined in RFC 4347 which is used as a tightly integrated, secure wrapper for a CAPWAP packet.
Nevertheless CAPWAP can be running in the same network together with LWAPP. The CAPWAP allows APs to join a controller that runs either CAPWAP or LWAPP. However, this is not the most recommended deployment as the "light" APs would have to download the specific firmware every time they join a controller, making the joining process longer.

Thanks to those features, CAPWAP is increasingly becoming the de facto standard for enabling operation between different hardware vendors.

2.2.3. Distributed LANs

Contrary to what centralized LANs offer, distributed LANs are based on distributing decision-making and control among network elements. This has been in part motivated by the rapid spread of technology that allows autonomy and decision capacity for every element to be increased. It also avoids the most serious drawback of centralized networks: single point of failure.

According to Mansoor et al. [13], distributed management techniques improve network availability and reliability as well as performance in comparison to centralized ones. In addition, the article states that it helps in increasing network capacities and improves its performance especially in large-scale WLANs.

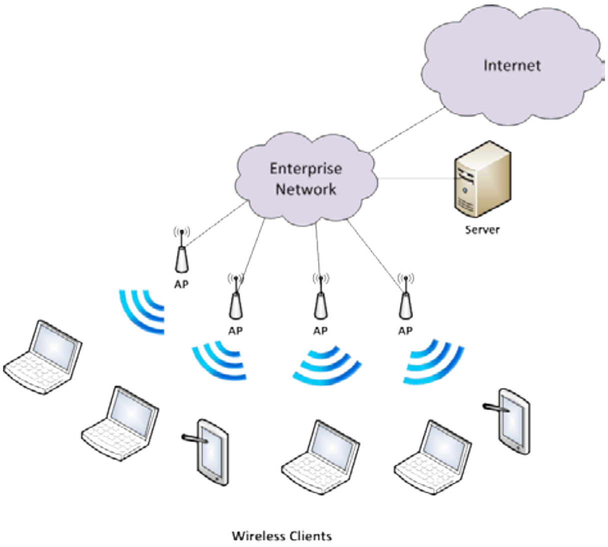


Fig. 2.8. Distributed WLAN architecture

Not only between APs and final clients, but sometimes communication between APs and the controller is also needed. This situation occurs when a company with a distributed architecture decides to integrate monitoring in a centralized way. In such cases, local autonomy is combined with central managed capabilities. Consequently, a balanced model between central management and local autonomy appears.

2.3. State of the art

In the literature, several techniques have been proposed to address performance issues such as excessive interference which usually translates into low throughputs. In addition to other radio resource management techniques, such as load balancing [14] and power control [15], channel assignment is extensively considered as a key strategy that should be taken into account and has thus been widely studied and applied. Basically, channel assignment in WLANs consists in assigning an optimal frequency channel to each AP for certain duration of time. Considering a WLAN consisting of a set of APs with its corresponding wireless clients, Chieochan et al. define channel assignment [16] as a strategy in which every one of the available channels is allocated to each AP such that the interference generated as a result is minimized.

In principle, it could be thought that common techniques used in cellular mobile systems could be also applied to WLANs, since WLANs can also be understood as a particular case of cellular network. However that is not a feasible possibility due to several particular characteristics of WLAN technologies. The most remarkable one is co-channel interference, whose effect in other cellular systems is entirely different due to the nature of the CSMA protocol used in WLAN. Other reasons are radio coverage and channel assignment techniques. About radio coverage, it is typically well planned in cellular network but such regularity is not present in WLANs, since WLANs use unlicensed spectrum. In relation to channel assignment techniques, they mainly differ in that WLANs use a single channel for data and control signalling, but cellular networks carry both types of traffic in different channels.

2.3.1. Related work

As we will discuss later, channel assignments for WLAN have become an interesting challenge and a wide range of research has tried to address the problem from different points of view. Below in this chapter, the main related approaches that can be found on the literature are discussed.

Aspects of a channel assignment scheme can be in general divided according to its nature or how often it is applied (static or adaptive), the type of deployment where it is applicable (uncoordinated or centrally managed), the type of frequency channel it uses (overlapping or non-overlapping channels), the procedure for obtaining channel assignment solutions (graph coloring, integer linear programming, bio-inspired algorithms) and from which perspective interference is studied (AP’s or client’s point of view). For the purpose of this thesis, focused on large-scale public access deployments, the most relevant papers are those treating aspects such as overlapping channels [17][18][19], adaptive nature[20] and AP-driven [21][20] [22] [23].

An interesting paper is [20] since it is based on the same management protocol the present work deals with. Its channel re-configuration system is performed via SNMP. Thus, several parameters obtained from the APs' MIB are combined so the performance at each one of them can be objectively measured and compared with neighbour results. In particular, MIB objects used in [20] are: the number of frames with errors (FCS), the number of frames delivered correctly (InUPkt), the number of associated stations (NAS) and the frequency channel. In this solution, a Java-based user-space software is installed on every AP, so they become intelligent and capable of exchanging information in a multi-provider Hotspot environment. This model however, has only been tested on a simulator and on a small testbed.

What [23] proposes is a dynamic radio resource management (RRM) technique that increases network performance and also reduces the co-channel interference. It works in such a way that each AP can be dynamically assigned to an optimal channel without the need of prior knowledge about network status. As a result, the overall network performance increases. Even so, this heuristic algorithm should still improve some issues related to the implementation of the technique. For example, authors intend to develop self-learning algorithms instead of tuning some parameters (accuracy-promptness, sensitiveness-frequency) which provide flexibility. This approach also lacks a procedure to inform associated stations about the decision of moving the frequency in use, what may imply the development of a signalling protocol.

Strongly related to this thesis is [18] the conclusions of which justify the use of partially overlapped channels in this work. In the paper, the Wireless Network Group of UPC presents an analytical study of the effects of adjacent channel interference (ACI) in IEEE 802.11abg WLANs. With the aim of evaluating the interference caused by transmissions in overlapping channels, they propose a model consisting in computing the power spectral density (PSD) of the filtered signal. This model which also quantifies the node utilization presents accurate results and concludes that partially overlapped channels are a useful resort when the number of non-overlapping channels available is small.

Despite the strong connection between this work and [18], there are also articles which deal with the downside of using partially overlapped channels. In [24], authors introduce an ACI factor when calculating SINR in a scenario with partially overlapped channels. This paper addresses the issue of ACI in 802.11a and reveals that it indeed exists although 802.11a standard was supposed to mitigate interference as a result of a better channelization combined with the use of OFDM transmission. Authors quantify the throughput degradation due to ACI both in data reception and clear channel assessment. When dealing with a channelization scheme using partially overlapped channels, authors state that some properties such as interchannel spectral distance, channel bandwidth, spectral mask and receiver filter get affected. To take all this into account, they introduce an ACI factor $X_{i,rx}$ for each of the interferers when calculating SINR. This factor depends on the spectral properties of the channels and the transmitted signals, as well as the separation between the channels of an interferer *i* and the receiver *rx*.

Some other papers study the same issue from another point of view. In [19] authors present a novel interference model that considers both the adjacent channel separation and the physical distance of the two nodes employing adjacent channels. By defining a node orthogonality model, they propose an approximate algorithm termed Minimum Interference for Channel Allocation (MICA). It is composed of three parts: (1) obtaining the fractional solution by solving a relaxed optimization, (2) obtaining the integral solution by rounding a process and (3) assignment of channels to APs. Hence, interference mitigation is studied through this algorithm which takes into account both the channel separation and the physical distance separation of the two nodes. The weakness this system presents can be mostly found in very dynamic environments, where neither the distance nor the positions of the neighboring APs are known parameters.

Following also the idea of considering overlapping channels, [17] focuses of the problem of efficient management of wireless spectrum based on dynamic measurements such as SINR model. Through this model, they formally deduced the conclusion that minimizing system interference is equal to maximizing the total network throughput. Therefore, a heuristic algorithm is proposed and several simulations are carried out with the aim of improving the performance of 802.11 wireless networks.

Dynamic measurements such as SINR are also used in [22] in order to characterize interference. The model they propose to accurately characterize interference in a wireless network and then make the right channel decision, incorporate dynamic real-world signal and also traffic measurements. Once interference is estimated, they propose a channel management algorithm which assigns weights on a graph so that channel reuse opportunities are maximally exploited.

In a recent research paper [25], authors present a channel assignment algorithm considering also adjacent channel interference (ACI) between non-overlapping channels, following ideas presented in [26] and [27] where it is explained how interference has a significant impact on overall throughput decline. Its proposed ACI aware channel assignment scheme uses an algorithm that recognizes the difference between co-channel interference and ACI which is reflected on a vertex coloring graph.

The approach presented in this work is based on [28]. In brief, APs exchange a collection of statistics following either a distributed or a centralized approach. Those statistics are the basis to build a “weighted graph” which is colored with a modified version of DSATUR algorithm. DSATUR algorithm is based on the concept of saturation degree and it uses a vertex coloring based approach. In a graph coloring, APs are treated as vertices of a graph, and a single edge of the graph represents potential interference between a pair of neighbouring APs.

The mechanism takes both co-channel and adjacent interference into account, and makes use of overlapped channels. This way, collisions as well as transmission errors are minimized, thus improving the network capacity and the user experience. More details on this mechanism are given in CHAPTER 4.

CHAPTER 3. DESCRIPTION OF THE ENVIRONMENT

This chapter is focused in the environment where the study takes place and the main characteristics it has from different points of view. Firstly, a description of the location of the campus is found, together with some significant data about the wireless network. Next, it addresses the description of those devices which compose the studied network as well as specifications regarding useful software for the purpose of this work.

3.1. Description of the environment

The emplacement where this study has been accomplished is in Universitat Politècnica de Catalunya (UPC) Barcelona Tech.

Among the twenty-three schools belonging to UPC, the two located in Campus Baix Llobregat (CBL) have been chosen for the tests carried out along this research work.

The CBL is home for ESAB (Escola Superior d’Agriculutra de Barcelona) and EETAC (Escola d'Enginyeria de Telecomunicació i Aeroespacial de Castelldefels), two schools where telecommunications and air navigation engineers, as well as agricultural engineers are instructed. Furthermore, in the Parc Mediterrani de la Tecnologia (PMT) there are also many research centres and companies. The image below shows an aerial view of the campus during its growing process.

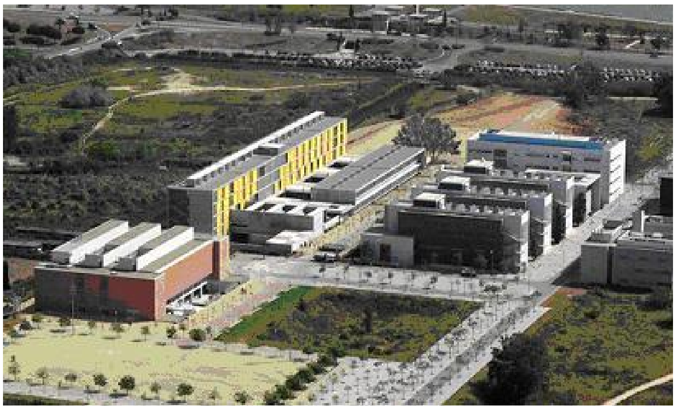


Fig. 3.1. Aerial view of the campus [2]

With the aim of providing Wi-Fi to the entire campus, there is a complex network of APs distributed around the campus area. They are controlled from a centralized point, the device which has provided us the information needed for the development of this study.

To be more accurate, the information presented along this report belongs to the services building from the campus, EETAC and ESAB schools, and three research centers: Agropolis, RDIT (Recerca, Desenvolupament i Innovació

Tecnològica) and CIMNE (Centre Internacional de Mètodes Numèrics a l'Enginyeria).
Providing Wi-Fi coverage to the total area of the campus represents a complex implantation of APs around the entire area. Moreover, in order to deliver Wi-Fi to any point of the campus, there is a large amount of neighbouring APs, and therefore, interference between them.

Fig. 3. 2 shows how APs see each other. As it can be observed, the map created from the neighboring relationship among APs results in a messy map with an innumerable quantity of links. This is the consequence of the aforementioned one hundred interfered APs which are connected with their neighbors through a sum of about five hundred links.

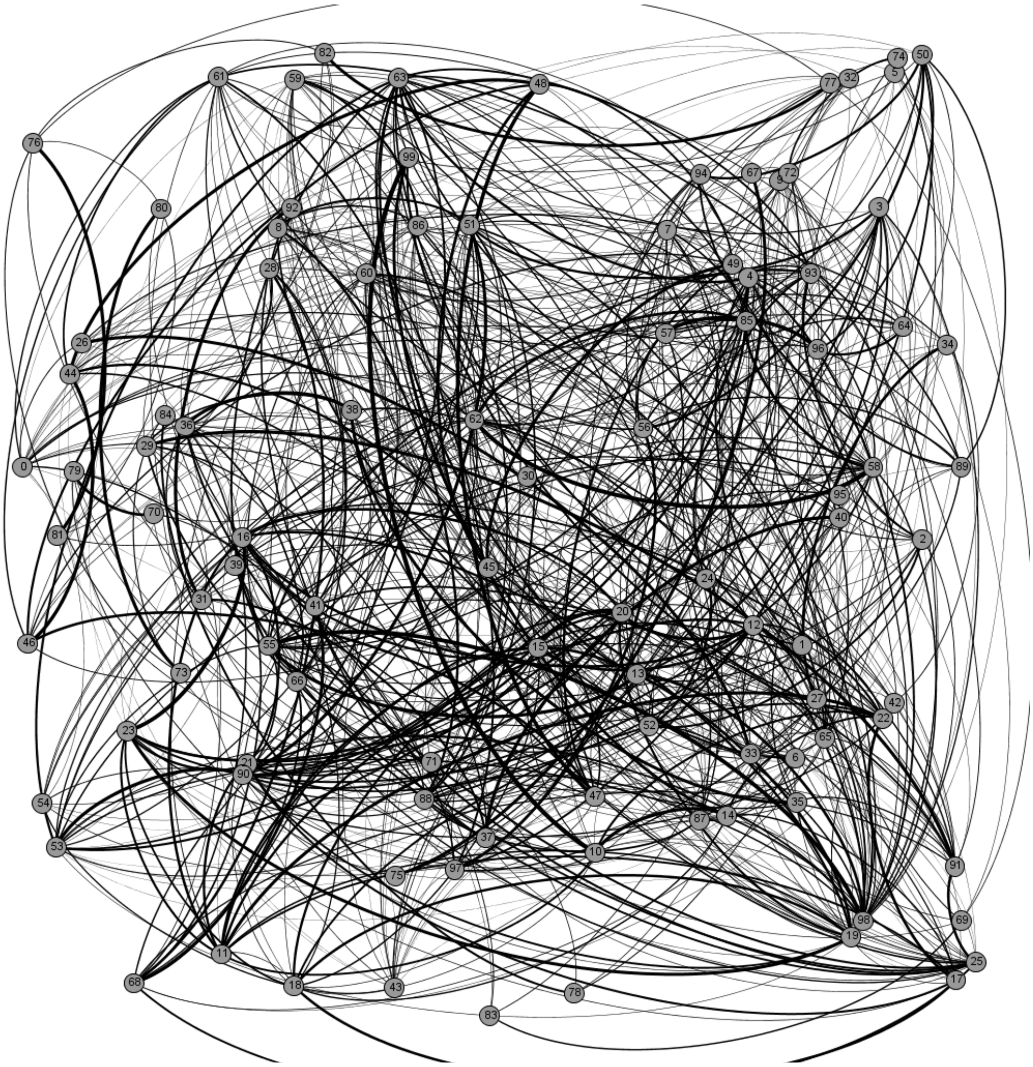


Fig. 3.2. AP graph of the analyzed APs

Not from this confusing graph but from the statistical analysis of all the information taken, we can say that every AP has, in average, 12.79 neighbour APs and 5.95 rogue neighbours, reaching a maximum of 26 neighbours seen by AP9; and 23 rogue APs seen by AP3. Both of them are located in the EETAC building.

It is important to emphasize the fact that, for the sake of clarity, the interference graph depicted in Fig. 3. 2 does not include any rogue AP, just those APs registered in the controller and hence, belonging to UPC.

Last but not least, another characteristic for the description of the environment we work with is its *clique number*. In graph theory, a clique defines a subset of nodes such that every pair of vertices is connected. As an example, Fig. 3.1 shows one clique of three nodes, {5,2,1}, and four others of two, {2,3}, {3,4}, {4,5} and {4,6}.

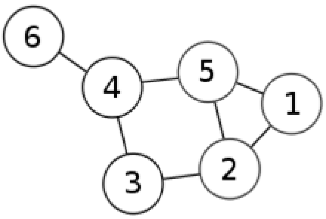


Fig. 3.3. Graph containing cliques

The clique number is the number of nodes in the largest clique found in a given graph. By analyzing the scenario with MACE_GO[1], we discovered 516 cliques containing from two to sixteen nodes. That is, the clique number for the interference graph representing our WLAN is 16, what means that we would need at least sixteen non-overlapping channels if we want to avoid interferences completely.

However, finding sixteen non-overlapping channels is not feasible at all given that the traditional frequency spectrum has just four non-overlapping channels at most (in the case of 802.11g). It is also interesting to note that most of the cliques are formed by five (119 cliques) or six nodes (98).

3.2. Hardware and software requirements

This section is intended to describe characteristics of those devices and software that play an important role along the development of this work. By describing them, the reader will easily make a further immersion into the implementation and the results, which are described in future chapters.

3.2.1. Cisco solution

Cisco Systems Inc.[29] is a multinational corporation born in the eighties in California (USA) for the designing, manufacturing and selling of networking equipment.

[1] MACE_GO: Maximal Clique Enumerator, ver. 2.0.

Fig. 3.4. Cisco Systems Inc.

Cisco is the trademark of the devices studied in the context of this thesis. An overview of the most relevant features is presented following in this section.

As previously mentioned in CHAPTER 2, Cisco Wireless LAN Controllers communicate with “controller-based” APs (or Lightweight APs) and it plays a central role in the network. Therefore, it assumes responsibilities traditionally carried out by the AP (e.g. association and authentication of wireless clients) as well as AP configuration and upgrading.
The specific model installed in the UPC premises for this purpose is Cisco 4404 Wireless LAN Controller (Fig. 3. 5) [30], specially designed for medium-to-large enterprises and capable of managing up to 100 APs. This model features four Gigabit Ethernet ports and two expansions slots to add an enhanced functionality and an optional redundant power supply. Table 3. 1 shows Cisco 4404’s most important features:

Table 3.1. Cisco 4404 Wireless LAN Controller specifications

Item	Specification
Wireless	IEEE 802.11a, 11b, 11g, 11d, 11h, and 802.11n
Security standards	IEEE 802.11i (WPA2, RSN) RFC 1321 MD5 Message-Digest Algorithm RFC 2246 TLS Protocol Version 1.0 RFC 2406 IPsec RFC 3280 Internet X.509 PKI Certificate and CRL Profile
Encryption	WEP and TKIP-MIC: RC4 40, 104 and 128 bits (both static and shared keys) SSL and TLS: RC4 128-bit and RSA 1024- and 2048-bit TAES: CCM, CCMP IPSec: DES-CBC, 3DES, AES-CBC
Authentication, Authorization and Accounting (AAA)	IEEE 802.1X RFC 2716 PPP EAP-TLS RFC 2865, 2866, 2867, 2869 RADIUS Web-based authentication
Management interfaces	Web-based: HTTP/HTTPS Command-line interface: Telnet, SSH, serial port

Fig. 3.5. Cisco 4404 Wireless LAN Controller

Regarding the APs connected to the controller, we can find three different models in the campus which are almost identical, but with minor differences between them. Cisco Aironet 1142N is the model most frequently installed around the site, but Cisco Aironet1231G and 1121G are also deployed.
All three models are designed for offices and similar environments, mainly because their built-in antennas, which provide omnidirectional coverage. Two versions are available: the *autonomous* one and the *unified* one; the latter is meant to operate together with a Cisco WLC. Since our environment follows a centralized architecture, the controller is supposed to manage unified LAPs. Nevertheless, we can also find some autonomous APs working for the controller. Fig. 3. 6 shows an image of the AIR-LAP1142N-E-K9, one of the most used models around UPC installations.

Fig. 3.6. AIR-LAP1142N-E-K9

The most remarkable characteristic that differentiates the three specific models is the radio card they include: while 1121G and 1231G models only work in the 2.4 GHz radio band, 1142N can also work in the 5GHz radio band. Different characteristics aside, the table below shows the most important characteristics for both controller-based and non-controller-based APs.

Table 3.2. Cisco AIR-AP1121G,AP1231G and AIR-LAP1142N specifications

	AP1121/31G	**LAP1142N**
Standards	IEEE 802.11b, 11g, 11i	IEEE 802.11a, 11g, 11n, 11i
Frequency band	2.4 GHz	2.4 GHz 5GHz
Data Rates	54 Mbps	Up to 300 Mbps @ 40MHz
Management	Autonomous	Controller-based

3.2.2. SNMP

The central controller manages operational and statistical information from every AP. By accessing the controller through SNMP we are able to extract all the information needed to analyze the network performance before and after applying our proposed channel management approach. A comparison of both results will allow the evaluation of the improvements in network performance.

Among the wide variety of SNMP information available, the analysis of MAC layer parameters will allow us to achieve the final goal. The tree represented in the figure below shows how MAC parameters are distributed in different branches.

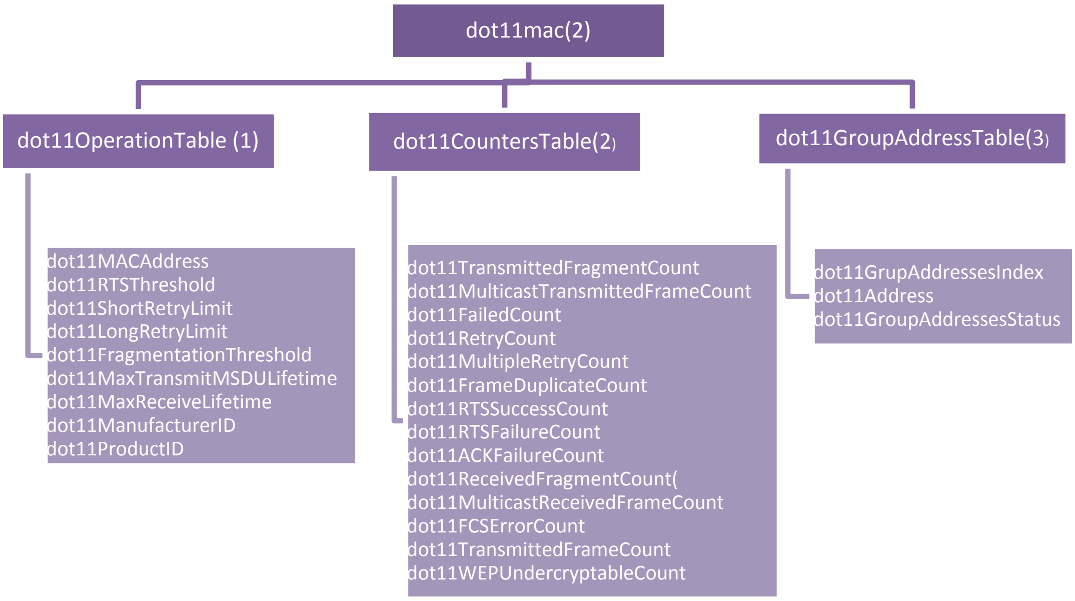


Fig. 3.7. SNMP parameters belonging to MAC counters

Parameters from Dot11mac MIB are specific to autonomous APs. However, the Cisco controller implements a more suitable MIB, the Airespace-Wireless-MIB [31]. It is intended to be implemented in central controllers and provides configuration and status information of managed APs.
This group of objects which represent the studied environment are located under the OID 1.3.6.1.4.14179.2[2], which includes twelve branches, each one devoted to a particular function. The figure below shows all these branches:

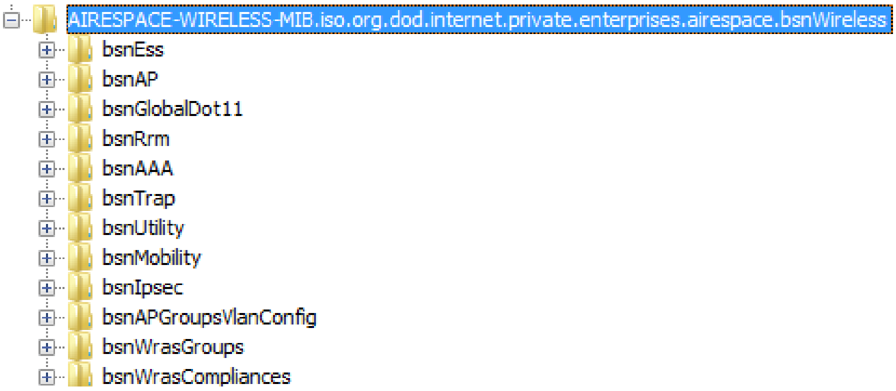


Fig. 3.8. SNMP parameters of a Cisco controller

[2] *iso.org.dod.internet.private.enterprises.airespace.bsnWireless*

Useful items for the search of optimum frequency channels are distributed between bsnEss(1) and bsnAP(2) branches. They contain objects associated to the configuration and operation of Wireless LAN (bsnEss) and information about configuration and operation of APs managed by the controller (bsnAP).
Within bsnAP we can find tables containing information about managed APs, as well as MAC counters and load parameters for every one of them. Among the information gathered, some of the most relevant parameters are:

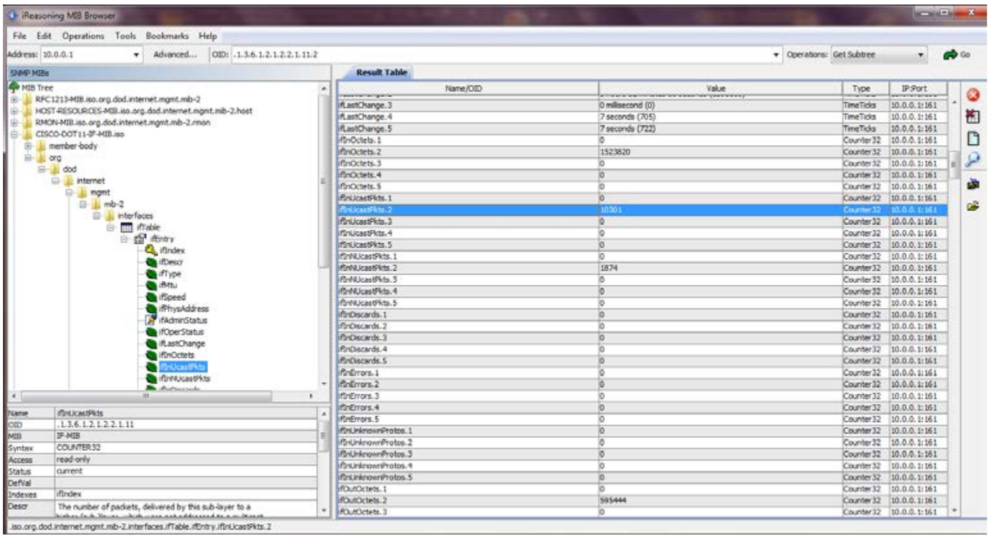
- *bsnAPIpAddress* (1.1.9)
- *bsnAPIfPhyChannelNumber* (2.1.4)
- *bsnAPIfLoadRxUtilization* (13.1.1)
- *bsnAPIfDot11TransmittedFragmentCount* (6.1.1)
- *bsnAPIfDot11RTSSuccessCount* (6.1.6).

Objects like *bsnRogueAPDot11MacAddress* (7.1.1), *bsnRogueAPChannel* (7.1.26) and *bsnRogueAirespaceAPRSSI* (8.1.7) have been examined to get a list of rogue APs. Rogue APs are those access points that do not belong to the managed network.

3.2.3. MIB Browser

A large variety of SNMP managers are currently available. For the development of the present work, iReasoning MIB Browser [32] and Net-SNMP [33] have been employed, as each one fits a different operating system.

iReasoning MIB Browser has different versions available on its website, free personal edition is the one used for the purpose of this work. This version has been used under Windows operating system. This tool stands out for its graphical interface, ease of use, IPv4 and IPv6 support, trap reception and sending, and support for both SNMPv1 and SNMPv2 protocols.



Address: 10.0.0.1
OID: .1.3.6.1.2.1.2.2.1.11.2
Operations: Get Subtree

Name/OID	Value	Type	IP:Port
ifLastChange.3	0 millisecond (0)	TimeTicks	10.0.0.1:161
ifLastChange.4	7 seconds (705)	TimeTicks	10.0.0.1:161
ifLastChange.5	7 seconds (722)	TimeTicks	10.0.0.1:161
ifInOctets.1	0	Counter32	10.0.0.1:161
ifInOctets.2	1523820	Counter32	10.0.0.1:161
ifInOctets.3	0	Counter32	10.0.0.1:161
ifInOctets.4	0	Counter32	10.0.0.1:161
ifInOctets.5	0	Counter32	10.0.0.1:161
ifInUcastPkts.1	0	Counter32	10.0.0.1:161
ifInUcastPkts.2	10301	Counter32	10.0.0.1:161
ifInUcastPkts.3	0	Counter32	10.0.0.1:161
ifInUcastPkts.4	0	Counter32	10.0.0.1:161
ifInUcastPkts.5	0	Counter32	10.0.0.1:161
ifInNUcastPkts.1	0	Counter32	10.0.0.1:161
ifInNUcastPkts.2	1874	Counter32	10.0.0.1:161
ifInNUcastPkts.3	0	Counter32	10.0.0.1:161
ifInNUcastPkts.4	0	Counter32	10.0.0.1:161
ifInNUcastPkts.5	0	Counter32	10.0.0.1:161
ifInDiscards.1	0	Counter32	10.0.0.1:161
ifInDiscards.2	0	Counter32	10.0.0.1:161
ifInDiscards.3	0	Counter32	10.0.0.1:161
ifInDiscards.4	0	Counter32	10.0.0.1:161
ifInDiscards.5	0	Counter32	10.0.0.1:161
ifInErrors.1	0	Counter32	10.0.0.1:161
ifInErrors.2	0	Counter32	10.0.0.1:161
ifInErrors.3	0	Counter32	10.0.0.1:161
ifInErrors.4	0	Counter32	10.0.0.1:161
ifInErrors.5	0	Counter32	10.0.0.1:161
ifInUnknownProtos.1	0	Counter32	10.0.0.1:161
ifInUnknownProtos.2	0	Counter32	10.0.0.1:161
ifInUnknownProtos.3	0	Counter32	10.0.0.1:161
ifInUnknownProtos.4	0	Counter32	10.0.0.1:161
ifInUnknownProtos.5	0	Counter32	10.0.0.1:161
ifOutOctets.1	0	Counter32	10.0.0.1:161
ifOutOctets.2	595444	Counter32	10.0.0.1:161
ifOutOctets.3	0	Counter32	10.0.0.1:161

Name	ifInUcastPkts
OID	.1.3.6.1.2.1.2.2.1.11
MIB	IF-MIB
Syntax	COUNTER32
Access	read-only
Status	current
DefVal	
Indexes	ifIndex
Descr	The number of packets, delivered by this sub-layer to a

.iso.org.dod.internet.mgmt.mib-2.interfaces.ifTable.ifEntry.ifInUcastPkts.2

Fig. 3.9. iReasoning MIB Browser

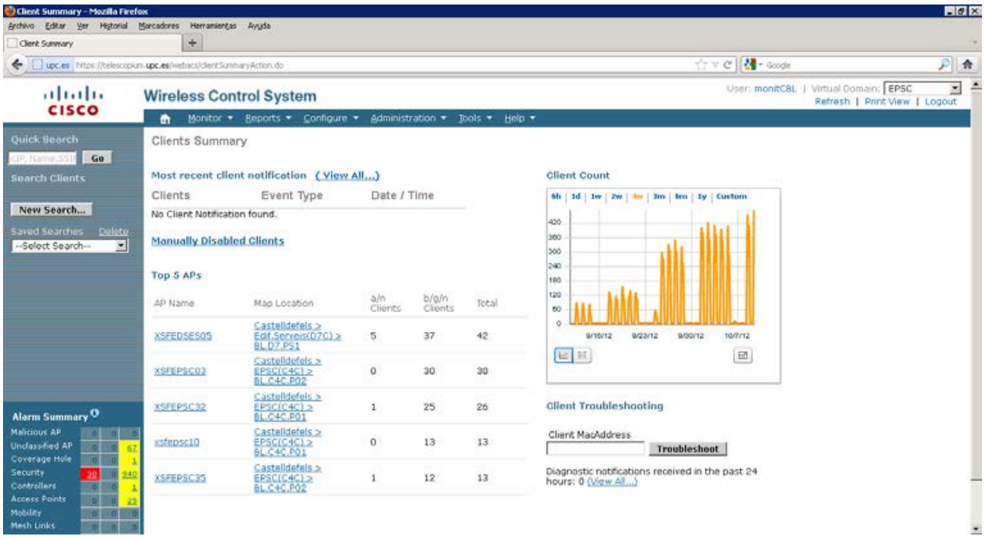
Net-SNMP [33] has been the SNMP environment used in the Linux platform used in this project. It is a Unix package for using and deploying SNMP

Protocol. It includes different SNMP tools: generic client library, command line applications, SNMP agent, among others.
For the communication with the network entity, Net-SNMP provides several commands. In spite of the variety, we must point out the following ones for their importance during the process: snmpget, which reads the value associated to a specific OID from the controller; snmpwalk, that uses sequential snmpgetnext requests until it receives all the information involving a tree or subtree; and snmpset, a command that allows writing values to a specific OID. Examples of them appear when implementation is explained in CHAPTER 4.

3.2.4. WLC's management interface

As previously mentioned, the management of all the studied APs is centralized in a single device, the Cisco controller. Even though the controller can be reached through different ways (e.g. Telnet or Secure Shell), this section describes the most remarkable features of its web interface, since this is the most friendly way given its graphical environment.

The web-browser interface (Fig. 3. 10) can be accessed by entering the controller's IP address in the browser's address line. It allows up to five simultaneous users configuring parameters and monitoring operation status of the controller and its associated APs.



User: monitCBL | Virtual Domain: EPSC

Clients Summary

Most recent client notification (View All...)

Clients	Event Type	Date / Time

No Client Notification found.

Manually Disabled Clients

Top 5 APs

AP Name	Map Location	a/n Clients	b/g/n Clients	Total
XSFEDSES05	Castelldefels > Edif.Serveis(D7C) > BL.D7.PS1	5	37	42
XSFEPSC03	Castelldefels > EPSC(C4C) > BL.C4C.P02	0	30	30
XSFEPSC32	Castelldefels > EPSC(C4C) > BL.C4C.P01	1	25	26
xsfepsc10	Castelldefels > EPSC(C4C) > BL.C4C.P01	0	13	13
XSFEPSC35	Castelldefels > EPSC(C4C) > BL.C4C.P02	1	12	13

Client MacAddress

Diagnostic notifications received in the past 24 hours: 0 (View All...)

Malicious AP	0	0	0
Unclassified AP	0	0	67
Coverage Hole	0	0	1
Security	20	0	240
Controllers	0	0	1
Access Points	0	0	23
Mobility	0	0	0
Mesh Links	0	0	0

Fig. 3.10. Cisco Wireless Controller GUI

The Graphical User Interface (GUI) allows the configuration, monitoring and control of the managed APs.
Among the monitoring functions, it is important to emphasize the detection of anomalies among which we can include rogue APs, and the possibility of scheduling the generation of different reports so that we can control some specific parameters. Additionally, it displays statistical information shown both in chart and in grid format.

3.3. XSF-UPC

In order to achieve the best implementation results, it is necessary to analyze the state of the network in critical moments. This includes detecting the most loaded APs and determining the rush hours, that is, the periods during which the load of the network is maximal.
It is known that the benefits of smart channel management are more clearly visible in moments of high load. In this section, we determine those periods after an accurate analysis of data.traffic analysis.

As mentioned before, the controller, through its graphical interface, offers statistical information shown both in chart and in grid format. The graph below is an example of it. In it, we can observe how users oscillate along a whole month during the academic year.

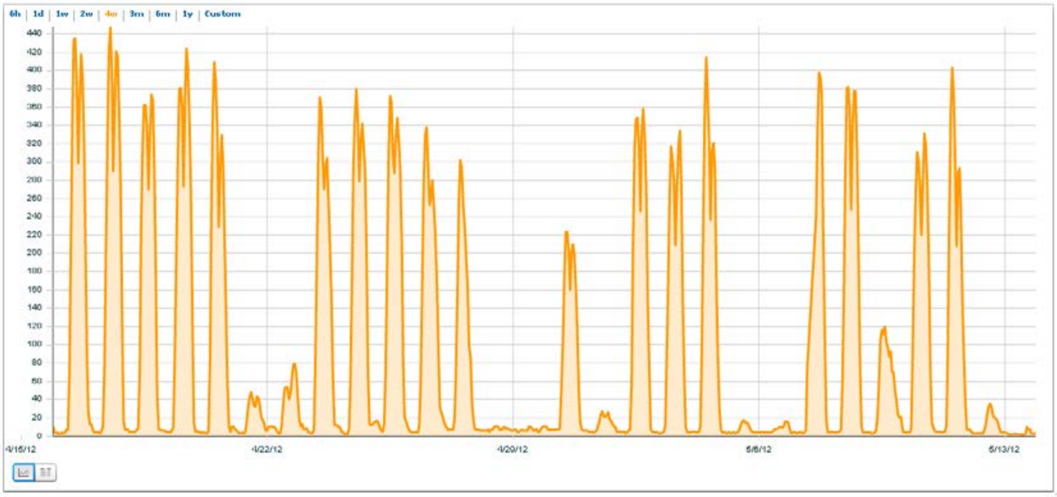


Fig. 3.11. Wi-Fi users along a month in CBL

The reader can observe that the pattern traffic is repeated at different scales. This scalar property in LAN network traffic is called “fractal” behavior. During the year there are months with more activity than others, coinciding minimum activity with holyday season. Then, at a lower scale, we observe that working days show more activity than weekends. Finally, at the lowest scale, this pattern is also repeated, showing peaks and valleys at different hours of the day.

Nevertheless, interesting moments for study are better extracted from link load graphs. Typically, link load is a useful tool for monitoring link status because it shows the total traffic generated along the time axis. This tool will be enough to guide us in the study of a typical user behavior in UPC Castelldefels.
In order to derive conclusions about the critical moments previously mentioned, we have to take into account users providing content (upload link) as well as users who are consuming it (download link). Fig. 3.12 presents these data extracted in May 2012.

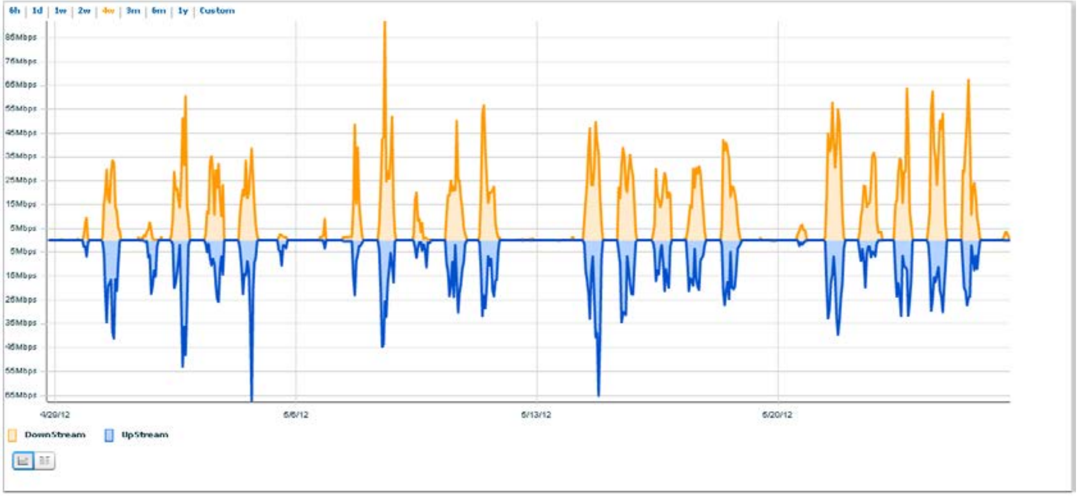


Fig. 3.12. Uplink and downlink traffic from Wi-Fi users along a month

The same graph but enlarged, that is, traffic information extracted from just one random day is presented in Fig. 3. 13, where we can observe that upstream and downstream graphs are almost symmetrical, and even peaks match in time.

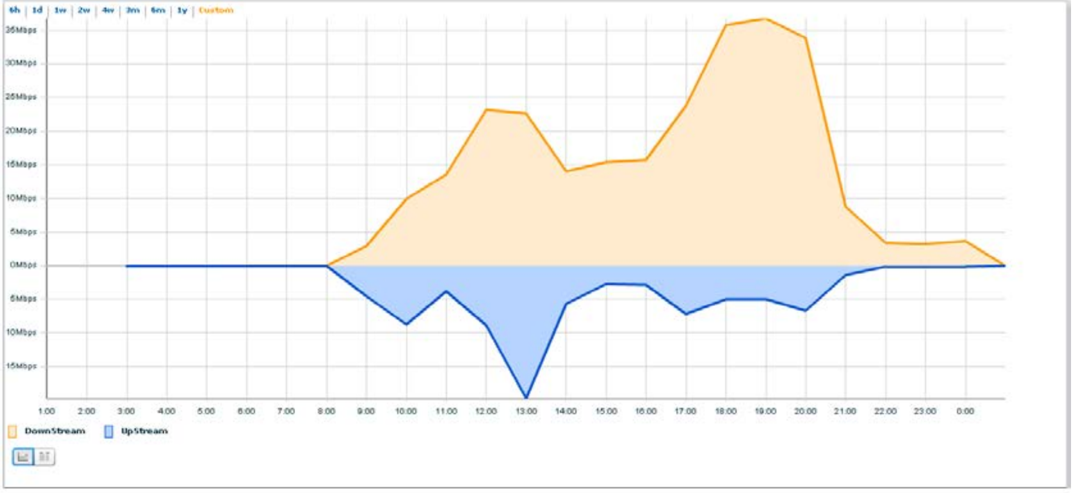


Fig. 3.13. Uplink and downlink traffic from Wi-Fi users in a random day

Besides the abovementioned symmetry of the graph, we can also observe two peaks, a pattern that is repeated along the time. Regardless of the day under study, there is a peak between 11:30 and 13:30 and a second one in the afternoon, between 18:00 and 20:00.
After the study presented above, we have concluded that information will be extracted during a month of normal activity, four times every working day from Monday to Friday at times matching the moments just before and after the observed peaks.

All the process followed for the obtainment and treatment of the SNMP information is carefully explained in CHAPTER 4.

CHAPTER 4. IMPLEMENTATION

This fourth chapter contains the explanation of the implementation itself. Besides describing the system designed through an example scenario, it first includes a full explanation of the approach chosen to perform an efficient frequency channel allocation.

4.1. DSATUR

This section is aimed to explain the details of the solution applied, so the reader will be able to have a detailed vision of the strategy used to perform the efficient radio resource management.

The channel assignment presented in this work consists on a mechanism that builds a "weighted graph" and proposes a frequency channel assignment. All of this, taking both co-channel and adjacent interference into account, and making use of overlapped channels. This approach is based on DSATUR algorithm [28], but overcomes existing limitations by adding some modifications which minimize interference between cells and consequently improves overall network performance.

Original DSATUR algorithm [34] is a sequential coloring algorithm that uses $G = (V, E)$ to model a given environment. This means that an interference graph G is formed by a set of "colored" vertices V interconnected by a set of edges E. Thus, V are considered to be WLAN cells in a region where E exists between two cells whenever they are overlapped. In graph coloring, APs are treated as vertices of a graph, and a single edge of the graph represents potential interference between a pair of neighbouring APs. The color assigned to a particular vertex is the lowest available channel, that is, the selected vertex is assigned the lowest channel not in use by any of its neighbours. Focusing on the relation established by an AP and its neighbours, the number of neighbours of an AP is called *degree*. *Saturation degree of an AP,* is *defined as* the number of colored neighbours it has.

At each iteration, DSATUR chooses the vertex to be colored [28][1] according to the following steps:

1. Arrange the vertices by decreasing order of degrees.
2. Color a vertex of maximal degree with color 1.
3. Choose a vertex with a maximal saturation degree. If there is an equality, choose the one with the highest ordinary degree (largest number of neighbours)
4. Color the chosen vertex with the lowest possible numbered color.
5. Return to step 3 unless all the vertices are colored.

The weak point it presents, however, is that it cannot always provide a feasible solution. When the number of non-overlapping channels available is not enough to avoid interference, DSATUR cannot provide a solution. It happens especially in highly interfered cells, where the use of overlapping channels causes performance degradation. To mitigate this effect, the approach goes through a more accurate interference characterization.

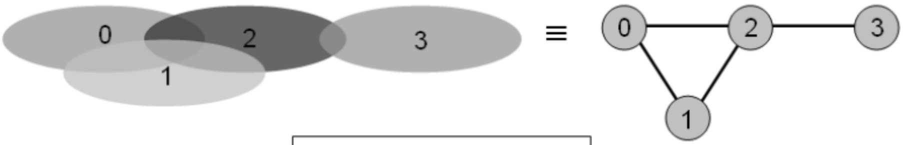
With the aim of fitting particularities of WLAN networks, some changes were proposed to the original DSATUR algorithm. The modified channel assignment algorithm includes some modifications regarding fast adaptation to changes and low degree of complexity. Therefore, the release is considered to be fast and to require few resources so it is suitable to be run in low-featured devices.

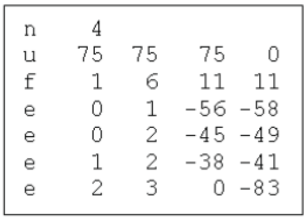
As said in [28][1], DSATUR does not work properly when there is a high density of nodes and edges. This happens because there are few available colors so a feasible solution cannot be found. This issue affects us directly as the most repeated clique in our scenario contains five nodes, so the traditional utilization of only three colors (channels 1, 6 and 11) would be inacceptable to obtain a good performance.

Besides these characteristics, the outstanding contribution of the approach lies in the fact that partially overlapping channels are included in the set of colors assigned. In this manner, both co-channel and adjacent interference are effectively reduced.

Also important is the contribution in terms of utilization each AP brings to the overall network performance. As a simple example, consider an AP which “sees” two APs, one of them is weak, in terms of received power, but it is always transmitting; the other one is a stronger interferer which seldom transmits, so it is not actually interfering. Hence, a smart channel allocation will try to avoid interference from the “weak”, but highly utilized neighbour. The lesson of this example is that not only RSSI but the sum of it together with utilization what indicates to what extent an interfering neighbour is harmful. Because of its importance, this DSATUR approach also adds utilization measurements to its channel assignment decision.

In order to run DSATUR, an interference graph combining information of all APs must be built. The text representation of the graph must include information such as the represented in Fig. 4. 1. It is obtained through requests to the controller (in a centralized solution) or the APs (in a distributed one), and includes *n* (number of nodes in the graph), *u* (utilization of each AP), *f* (the original frequency channel assigned to each AP), and *e* (representation of interference relationships between APs in terms of RSSI). A 0 in an interference *e* line position denotes that information is unknown).





```
n    4
u   75  75   75   0
f    1   6   11  11
e    0   1  -56 -58
e    0   2  -45 -49
e    1   2  -38 -41
e    2   3    0 -83
```

Fig. 4.1. An interference graph and its text representation [1]

More detailed information about the approach used can be found in [1], where it is exposed in detail.

4.2. Code explanation

This section contains the explanation of all the steps followed by our centralized channel management scheme. The whole process consists on several steps, which starts with the extraction of SNMP information and finishes with the modification of some parameters of the network.
With the aim of extracting more reliable conclusions, changes have been applied during a prolonged time. To do it periodically during several weeks, we have used the Unix job scheduler, Cron, which have allowed us to automate the programmed tasks.

4.2.1. Example scenario

Here an example scenario is presented. It will be a good way for the reader to get an idea of the complete process, as it is explained step-by-step. Its understanding will make it possible to get a sense of the entire network picture and the total amount of information processed. However, to a narrower comprehension of the process we recommend the reading of ANNEX A, where practical results and statistics are included.

Especially in ANNEX A, but also in this section, there are references to the simulated scenario described by Fig. 4. 2, which is composed of six APs (ID:0 to ID:5) and three rogue APs (ID:6 to ID:8) connected as follows:

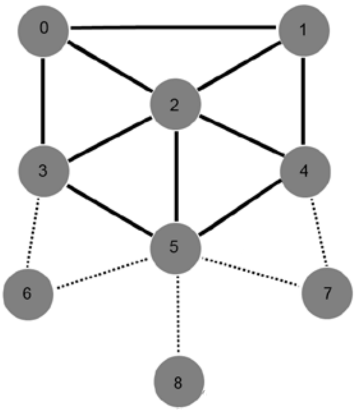


Fig. 4.2. Simulated scenario

The edge thickness represents the RSSI level, so that there is less interference between any neighbour AP than between any AP of the network and a rogue one. In other words, and to ease the comprehension of Fig. 4. 2, if all nodes worked at the same frequency channel, AP4 would receive interfering signal from AP1, AP2, AP5 and AP7; however, the one received from AP7 would be weaker, as represented by the dotted line. AP1 or AP3 will never interfere AP4, regardless of the frequency channel chosen.

4.2.2. Getting SNMP information

In order to get that SNMP information interesting for what concerns us, we need to know which OIDs provide it. For this purpose, and also with the aim of knowing which SNMP parameters are available, an analysis of SNMP packets via its MIB Browser GUI has been accomplished.

Once we differentiate those OIDs which include the information needed, it is time of obtaining the SNMP information through a series of commands similar to the following one:

```
snmpwalk -v2c -On -c **password** **controllerIP** 1.3.6.1.4.1.14179.2.2.1.1
```

To see a text file with the result of the different *snmpwalk* requests referring our test scenario, see Annex 1. There, the reader can observe that an AP managed by the central controller, contains plenty of information, such as its MAC address, frequency channel, RX, TX and channel utilization, MAC counters, AP neighbors and the RSSI with respect to its neighbors. However, this is not what happens when we study rogue APs. In that case, the controller is only able to obtain information regarding rogue APs' MAC address, frequency channel and the RSSI between the managed and the rogue AP.

4.2.3. SNMP data analysis

With the aim of transforming the data obtained into interesting information for us, we have developed a Java program which analyzes daily SNMPwalks and extracts statistical information as well as those input files needed for further phases.

The Java project is basically composed of two main objects:

- *AP*. It is formed by those fields that uniquely identify an AP, i.e. *OID*, *ID*, *MAC*, *channel* and its *neighbour list*. Also its *load* and *MAC counters* are stored because of its importance for statistical analysis. Finally, it also has a field to fill with the *new channel* the algorithm assigns.
  These objects represent the APs controlled in a centralized way. Taking the scenario from Fig. 4. 2, APs whose ID goes from 0 to 5 belong to this group.
- *Neighbor*. An AP can interfere with both controlled APs and rogue APs, from which we do not have detailed information. A neighbor is identified with its *OID*, *ID*, *MAC*, *RSSI* and *times* (number of times it has been identified as neighbor of an AP).
  For the example scenario we present, this object represents any AP that is detected by at least one of the controlled APs: AP5 is neighbour of AP3, AP2 and AP4, AP0 is neighbour from AP1, AP2 and AP3, and so on. Within *NeighborAPs*, those APs not managed by the central controller (i.e.rogue APs) are also included, that is, APs whose ID goes from 6 to 8.

The whole process is formed by a set of chained functions; they are explained in order below:

```
man.initialdata();
```

At the same time that a *snmpwalk* is performed, an *AP list* and a *MAC list* are also created. They will allow keeping control of the APs which form the scenario.

With this starting data set, a list of APs is created. Within this phase, only basic information (OID, ID and MAC) is filled. However, the rest of information will be provided within further methods.

```
man.readfiles();
```

This function parses the daily logs and looks for information that fits with the gaps existing in the AP object created.

After running this function, every AP object will count with information concerning its RX, TX and channel load, as well as the neighbors it has and the signal strength received from each of them.

```
rogue.readfiles();
```

In the same manner, information related to rogue APs is analyzed and classified according to the AP which detected it. Furthermore, every new AP is added to the *AP list*, so that a general control of APs is taken.

To see the amount of information extracted from the example scenario *snmpwalk* at this point, see Annex 1.

4.2.4. DSATUR input and output

The decision about the new channel in which every AP should operate so that the global network performance gets improved is taken by DSATUR. By extracting the information in an appropriate format, this algorithm will provide a solution in which the interference between neighboring APs is effectively reduced. For more details about the algorithm, refer to section 4.1**¡Error! No se encuentra el origen de la referencia.**; there DSATUR algorithm as well as the text representation suitable for its input are exposed.

```
sta.map ();
```

Its goal is the generation of an input file suiting the DSATUR input format.

Refer to ANNEX A for self-explanatory examples of the input/output files corresponding to the scenario depicted in Fig. 4. 2.

4.2.5. Setting SNMP parameters

In the same way that *snmpget* command allowed us to obtain information, we will apply the appropriate changes through the *snmpset* command. We are particularly interested in two OIDs; first of all, we will set *PhyChannelAssignment* to customized, to apply then to the specific *ChannelNumber* assigned.

Thus, the result for any one of the APs will be something similar to the following lines:

```
snmpset -v2c -On -c **password** **controllerIP** *specificOID* i 2
snmpset -v2c -On -c **password** **controllerIP** *specificOID* i 6
```

To do it in an automatic way during all the period of time under study, a script is constructed taking the daily DSATUR output as a reference.

4.2.6. Quantifying the improvement

Once changes are set, there is the need of comparing the performance of the network before and after setting the new APs configuration. This way, we will have a way of quantifying how far the wireless network performance improves.

Ideally, throughput is thought to be the most appropriate performance metric, as it is directly connected to the Quality of Experience (QoE) of the user. However, in order to measure the available system throughput we would need the sum of all the data rates that are delivered to all terminals in the network. On this basis, throughput monitoring is not suitable given the characteristics this real scenario has, since we would need the totality of APs (more than 100 APs!) working at the highest performance, which is unfeasible.
Because of this reason, we made the decision of exploiting MAC counters, parameters available in the controller, and finding a most appropriate way of studying the network performance for this specific scenario.
Among the wide quantity of MAC counters available in the controller, we have determined that the most relevant ones are *TxFragmentCount*, *RetryCount*, *FCSErrorCount* and *RxFragmentCount* which belong all to bsnAPIfDot11CountersTable[3] subtree.

On the one hand, *TxFragmentCount* and *RetryCount* parameters will provide us information about upload error rate, so that:

$$1 - p = \frac{TxFragmentCount - RetryCount}{TxFragmentCount} \qquad (4.\ 1)$$

$$p = 1 - \left(\frac{TxFragmentCount - RetryCount}{TxFragmentCount}\right) \qquad (4.\ 2)$$

Download error rate will be given by the *FCSErrorCount* with respect to *FCSErrorCount* + *RxFragmentCount*.

[3] 1.3.6.1.4.1.14179.2.2.6.1.1

$$p = \frac{FCSErrorCount}{FCSErrorCount + RxFragmentCount} \tag{4. 3}$$

This way, formulas shown beyond will make possible the quantification of both uplink and downlink PER for every AP of the network. In short, every AP performance will be measured as a ratio (%) relating packet error count to the total number of packets tested.

CHAPTER 5. EVALUATION OF THE SYSTEM

After studying the network characteristics and designing a channel management system, it has been running during a three-week period. This chapter presents the analysis of the most relevant parameters during the time the system has been tested.

5.1. Initial considerations

In CHAPTER 3, a detailed study of uplink and downlink traffic is presented. From that analysis, we concluded that SNMP information should be extracted during a normal activity period, four times every working day, matching the moments just before and after the two daily peaks.

After that initial conclusion, we have followed the same guide during three weeks, which involved: getting SNMP information at times decided, processing it in order to get the appropriated channel distribution according to DSATUR algorithm and conducting changes at last time of the day. This way, the users could enjoy the new channel distribution the next working day. Meanwhile, SNMP information is extracted again and changes applied if necessary at night.

This chapter is intended to analyze the benefits of the proposed scheme, but before that, it is necessary to point out what characteristics would be considered as improvements due to the new channel assignment. As explained in CHAPTER 4, the most interesting parameters to assess the improvements of our approach are related to frame error probability. Frame errors are produced by different factors: low SNR (e.g. due to distant clients) and collisions due to highly loaded APs and interfering cells. We henceforth assume that the distribution of users was the same during the different stages of the project, that is, the number of errors due to low SNR and loaded APs is similar before and after applying our mechanism. Therefore, the differences observed in those parameters are due to changes in the number of collisions after applying our interference-aware frequency management system.

Last but not least, let us remember that in CHAPTER 4 PER tracking was said to be the most appropriate way of quantifying whether changes applied bring improvement to the overall network performance or not.

Although this was the original idea, we have encountered a bug[4] in the Cisco WLAN Controller software release that hampers the computation of PER from some MAC counters, which are reported to display incorrect values.

Considering which statistical parameters we can rely on, we have chosen the option of comparing *FailedCount*, *MultipleRetryCount* and *FCSErrorCount*. All of them seem unaffected by the bug and allow us to make an evaluation of our scheme. Through these measures, however, we are not able to know exactly

[4] http://www.cisco.com/en/US/docs/wireless/controller/release/notes/crn52xgmr1.html

how much adjacent (related to PER) neither co-channel interference (via collision probability) evolve.

5.2. Frequency channels assignment

Since the key point of this system is to provide an efficient channel allocation, it is imperative to examine how frequency channels have been distributed during the period the system has been running.

Before going into details about the performance of our channel assignment strategy, let us remember how the average channel distribution was during the observation period (Fig. 5. 1).

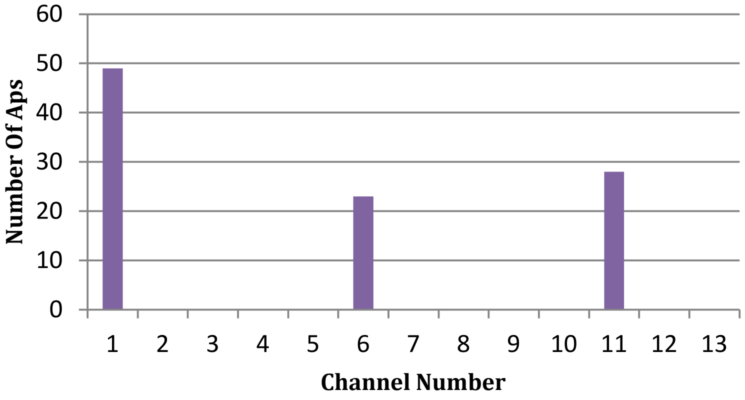


Fig. 5.1. Starting channel distribution

As can be seen in these statistical data extracted before running our system, all APs were originally working in the traditional three channels 1, 6 and 11. It is important to note that almost half of them were set to work in the first available channel (i.e. channel one). This channel distribution is the one chosen by Cisco's auto-channel selection algorithm, which scans all the available channels and selects the lowest unused frequency for each AP. It is worth to mention that several comments on Cisco forums[5] recommend a static frequency assignment.

Fig. 5. 2 was created using data available both before and after applying our settings. It summarizes how channels were distributed before (first bar) and how they have been assigned according to DSATUR algorithm decisions.

As we had already foretold, DSATUR performance should make the allocation to be organized differently, according to all the factors it takes into account and the entire range of frequencies it considers when deciding the most efficient channel allocation.

[5]http://www.cisco.com/en/US/docs/wireless/access_point/12.4_10b_JA/configuration/guide/scg12410b-chap6-radio.html

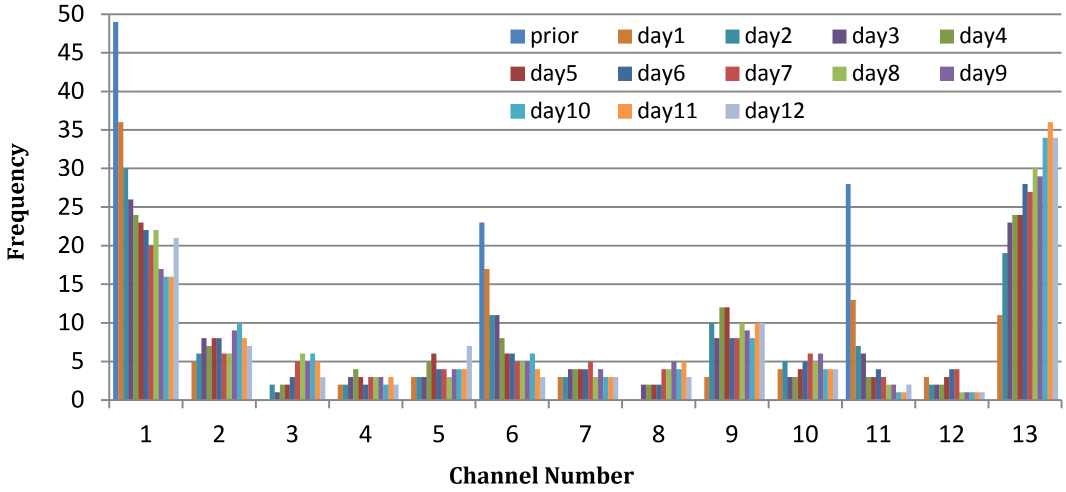


Fig. 5.2. Channel assignment along time

In fact, we can appreciate how the number of APs set in channels 1, 6 and 11 changes gradually so that, at the end, there are as many APs set in the traditional channel frequencies as in the rest of the range. This gradual change is due to the environment characteristics, since DSATUR takes into account which neighbours and rogue APs each AP sees as well as their utilization and the channel they occupy. Likewise, we can guess that rogue APs may have changed their working frequency channel, and so, the incoming channel allocation has been adapted to changes. To know how DSATUR has daily allocated channels in the managed network, see Annex 2. There, a larger version of Fig. 5. 2 is also included.

As seen in Fig. 5. 2, there is a gradual change tending to relocate APs' working channels so that they became almost equally distributed. To measure it, there is an index called Jain's index[35] which is well known for providing a fairness measure on the distribution of resources (i.e. channels) among consumers (i.e. APs). Hence, it is very used in network engineering to determine if system resources are being correctly distributed. Jain's index is expressed through the following formula:

$$\mathcal{J}(x) = \frac{\left(\sum_{i=1}^{n} x_i\right)^2}{n \cdot \sum_{i=1}^{n} x_i^2} \qquad (5.\ 1)$$

From its definition we can say that this is a continuous function and has a range $\left[\frac{1}{n}, 1\right]$, being its minimum value when only one user receives a non-zero benefit (least fair allocation) and its maximum when all the users receive the same benefit (the fairest allocation). This formula has been applied for the data collected from every day, resulting in a graph like Fig. 5. 3:

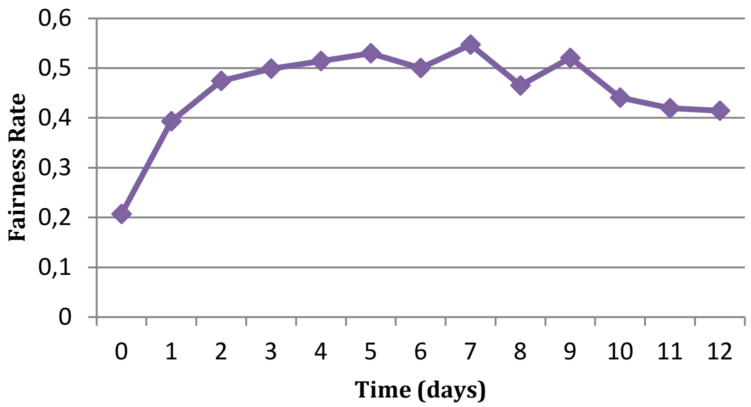


Fig. 5.3. Jain's fairness index

This graph describes somewhat the daily changes seen in Fig. 5. 2. It starts with a low fairness index (0.2) which belongs to Cisco's default configuration. As seen in the figure, the first day after applying the new configuration this rate increase up to 0.4. From that moment, it describes a line which grows slowly until it reaches 0.53 (fifth day). Later on, this metric does not follow a clear trend, but in general we can establish that it ranges between 0.45 and 0.55, depending on the network characteristics.
A question arising might be why this value is not higher. Then, we have to take into account that Fig. 5. 3 just describes the channel allocation of those APs managed by the controller. Recall that DSATUR algorithm decides daily channel distribution subject to the interference "seen" by each AP, which includes rogue APs.

In Fig. 5. 4 we can better distinguish how the distribution of channels remains after the changes applied by the system. In particular, the information presented in the figure belongs to the last testing day.

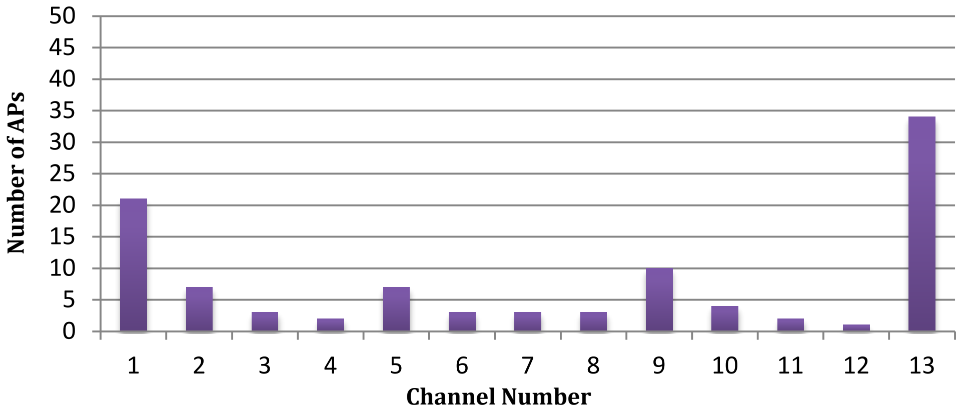


Fig. 5.4. Final channel distribution

We can observe that after running the algorithm during a prolonged time, APs are working in all the available channel space (from 1 to 13). Also we can appreciate that most APs are working either in channel 1 or 13, that is, the channels located at the ends of the range. This fact has a simple explanation since these are frequencies more likely to be chosen by the algorithm because they suffer from less adjacent channel interference.

Although to a lesser extent, we can see peaks in channels 5 and 9 (besides 1 and 13, already mentioned). This explains what [18] sustains, the idea of using the thirteen available channels, sacrificing channel spacing and so, having available the equidistant 1, 5, 9 and 13 channels.

About the difference in the number of APs set in channels 1 and 13, we have already mentioned, Cisco APs tend to configure itself to work on channel 1. Then, we can also expect many rogue APs to be also set in channel 1. For this reason, in the new channel distribution we find a lower number of APs working on the first channel in comparison with 13, which would be otherwise unused.

5.3. MAC counters

In order to evaluate the goodness of the system, this section is intended to compare how some MAC counters have varied with respect to Cisco's default working mode. This way, we will be able to provide values of how the presented system contributes to enhance the wireless network performance of the campus.

As a first approximation, Table 5.1 presents global statistics of the effects arising from the new channel assignment. To do it, we have averaged the ten highest values of each counter (i.e. the ten worst APs) over near one month of operation, from both before and after running our system, and thereupon, compare them. The comparison (third column of Table 5.1) presents the relative increase/decrease rate of the given counters with respect to the values obtained under default operation.

As we can see, the new configuration contributes to have a decrease in *FailedCount* and *FCSErrorCount* counters (of 74.34% and 40.90% respectively) and an increase of 26.66% in *MultipleRetryCount* counter.

Table 5.1. MAC counters global statistics

MAC Counter	Reference Data	Proposed Approach	%Increase
FailedCount	6,600.89	1,694.36	**-74.34**
FCSErrorCount	42,322,730.39	25,011,549.72	**-40.90**
MultipleRetryCount	1,049,839.76	1,329,782.72	**26.66**

If we pay attention to absolute values we realize that there is a large variation in the order of magnitude of the values. To understand it, we need to know the different connotations every MAC counter has according to the description provided by its MIB:

- *FailedCount* counter increments when a MAC Service Data Unit (MSDU) is not transmitted successfully after several attempts. Depending on the frame size, the number of attempts can be designated by the *ShortRetryLimit* (7) or the *LongRetryLimit* (4).
- *FCSErrorCount* counter increments its value when an FCS error is detected in a received MPDU.

- *MulipleRetryCount* counter increments when a MSDU is successfully transmitted after more than one retransmission.

Keeping to these definitions, it is reasonable that *FCSErrorCount* presents the highest absolute values, as it considers every reception failure. We also note that *MultipleRetryCount* is considerably higher than *FailedCount;* that is because the first one considers those packets which needed between two and the allowed maximum number of retransmissions, while the second one only considers those packets that failed after the allowed number of retransmissions (limited by *ShortRetryLimit* or *LongRetryLimit*).

Focusing on the third column, which describes the improvement achieved, we observe that *FailedCount* has considerably decreased its value, providing one of the reasons explaining why *MultipleRetryCount* was incremented. Also *FCSErrorCount* has decreased its value. More discussion on the evolution of these parameters is found in the following analysis, where daily statistics are compared to the averaged result of the Cisco System's default behavior. As in the global calculations, values belong just to those APs having the ten highest values.

Anyway, through Table 5.1 we can already advance that changes applied assure benefits in the performance of the managed network.

5.3.1. FailedCount

Firstly, let us compare *FailedCount* counter (see Fig. 5. 5) which is maybe the most critical counter since an increase of its value means that the transmission of a packet has failed despite several attempts. When it happens, it is also considered as a packet loss by the upper layers, thus affecting their performance. Depending on the working mode of the upper layers (e.g. in TCP protocol), this can mislead the congestion control mechanisms which aim to minimize losses but affects the performance of the network experienced by the end user when losses are due to channel errors instead of congestion.

The number of frames that are dropped due to excessive retries depend on the PER of the network. More precisely, *FailedCount* could be described by the monotonically increasing function shown below.

$$P_{FC} = PER^5 \tag{5. 2}$$

It represents the probability that a packet fails five consecutive times and its value will decrease as long as PER decreases.

Then, on the basis that our channel assignment reduces packet error rate, we expect to have a decrease in the value of this counter. The result is found in Fig. 5. 5.

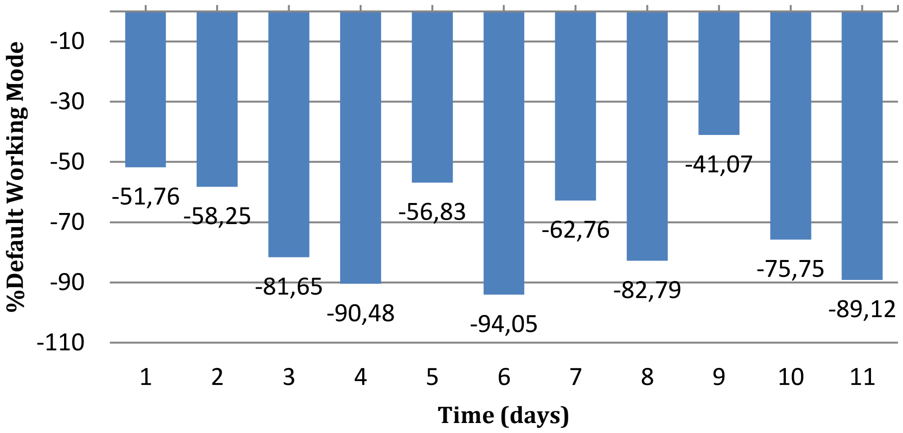


Fig. 5.5. FailedCount counter progression

In Fig. 5. 5 one can observe that the counter has always decreased its value in comparison with reference data.

Depending on the day, we find that there are between a 40% and a 94% less data units which could not be sent, in other words, more data units reach their destination with no intervention from upper layers.

Of course, the rate of improvement oscillates day by day, as this is a real environment and many factors are unpredictable to the skills of the system. In any case, these results represent an excellent performance improvement in the managed network.

5.3.2. FCSErrorCount

FCSErrorCount increments its value when a Frame Check Sequence (FCS) error is detected after the reception of a frame and consequently, it must be retransmitted. If we compare its behavior along the days, we get the data collected in Fig. 5. 6.

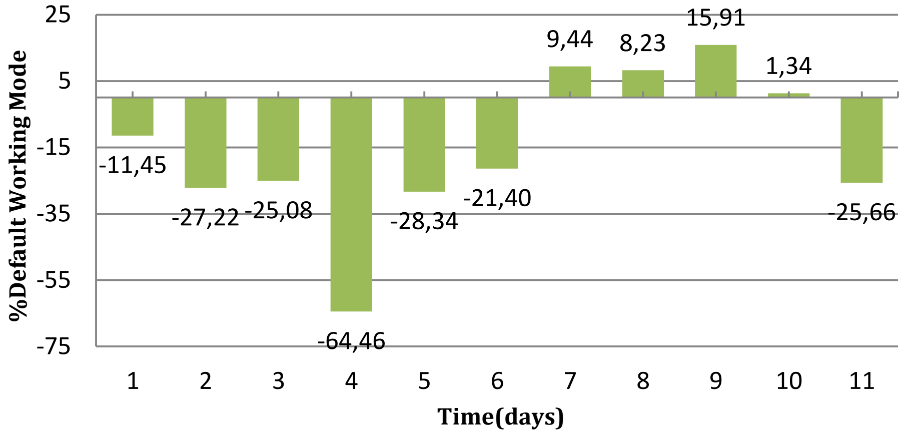


Fig. 5.6. FCSErrorCount counter progression

Unlike *FailedCount* counter, this one fluctuates between positive and negative values, providing improvements some days but deteriorations some others (always with respect to the reference data). Wireless networks deal with a high level of uncertainty since wireless signals are affected by unpredictable factors, phenomena that could explain the variations between the days the system has been tested. Since the trend this counter follows is not clear at all, we would need to test the system during more days in order to get more information and be able to extract more reliable conclusions. Even so, we can see (Fig. 5. 6) that most of the time the network shows a satisfactory performance, result of our channel assignment approach.

5.3.3. MultipleRetryCount

We finally have *MultipleRetryCount*, the only counter which increments its value with respect to the prior channel configuration scheme.

As we have already anticipated, this counter is slightly connected to *FailedCount*. After reducing the interference, the packet error rate is decreased and hence an improvement in the number of frames which fail to be transmitted is expected. Initially, one could thing that this would also reduce the number of frames that required several retransmissions to reach their destination. Hence, assuming similar offered traffic during the two stages observed (Cisco's normal operation and our channel management), we could expect a decrease of the *MultipleRetryCount* counter in response to the decreased *FailedCount*.

However, according to the statistics collected in Fig. 5. 7, we find that in 90% of cases this new channel distribution leads to a considerable increase in the number of frames successfully transmitted after several attempts. This behaviour is explained when analyzing the following function:

$$P_{MRC} = (1 - PER) \cdot \sum_{i=2}^{4} PER^{i} \qquad (5.\ 3)$$

This function represents how *MultipleRetryCount* evolves according to the packet error rate, and takes into account retransmissions between two and the retry limit (which is four in the case of data packets).

Unlike *FailedCount*, this function increases up to a maximum point and then decreases. Thus, *MultipleRetryCount* value can increase or decrease depending on the value that PER takes.

As we see in Fig. 5. 7, most of the days a reduction of the PER involves an increase in *MultipleRetryCount*. Besides, we see that one of the days differs, that is, *MultipleRetryCount* is decreased with PER.

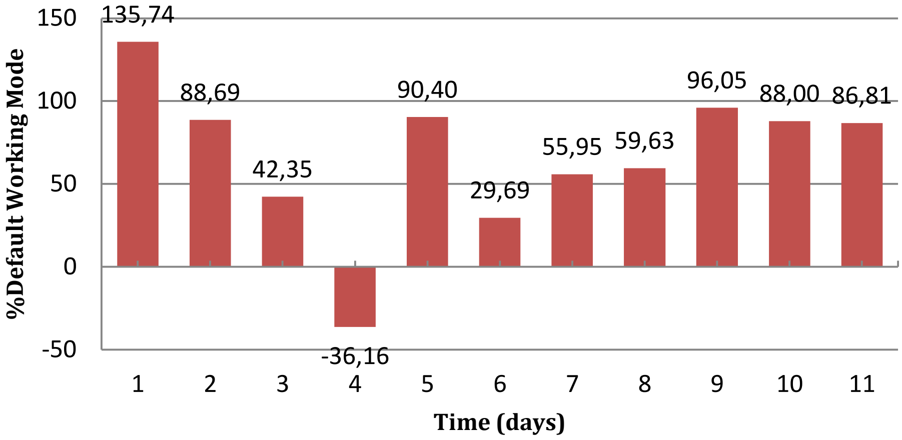


Fig. 5.7. MultipleRetryCount counter progression

5.3.4. Evolution of MAC counters

Besides the previous individual analysis, it can be also a good idea to merge the already discussed graphs in a single one. The evolution of all three counters is shown in the following graph (Fig. 5. 8).

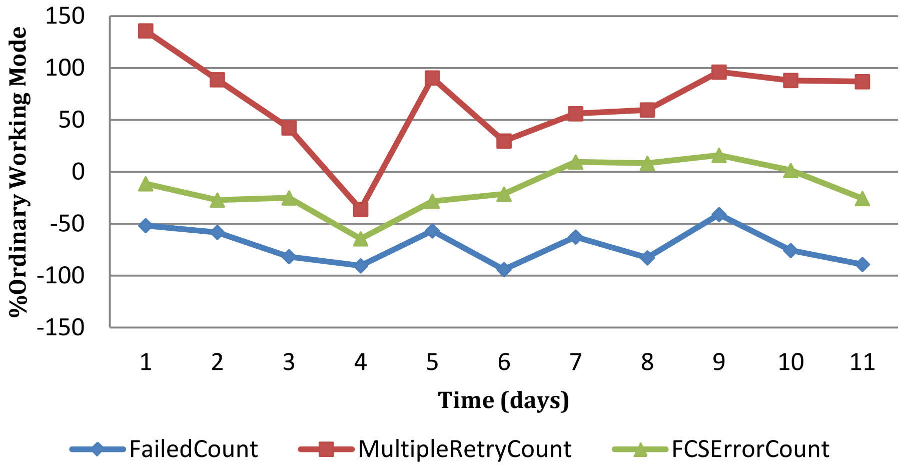


Fig. 5.8. MAC counters progression

In this graph where *FailedCount*, *MultipleRetryCount* and *FCSErrorCount* are displayed, we can observe the general progress of every MAC counter in relation with the others.

Just at first glance we see that *MultipleRetryCount* stands out over the other counters because of the high range it occupies. All counters start with a high value; blame of the first channel assignment set, as it was the result of processing the average information over all the observation stage, that is, very old data was used for the first assignment. Another outstanding point is the drop referred to the fourth day. This fourth sample corresponds to the last working day before Christmas holidays. Hence, many people had already started their holidays and the traffic load of the network became considerably lesser. Under these conditions, the medium is considered to be less loaded (less interference and fewer collisions will be produced) and, in consequence, this will cause a decrease in PER rate. When PER is low, a decrease in the number of frames requiring retransmission is observed.

We can see next that once the holiday period finishes, values of the three counters are back within the average range.

About the behavior of *FailedCount*, it is remarkable the great improvement it implies. Indeed, its lower value (-41.07%) represents almost half of the frames fail in relation to the default operation, even achieving the delivery of almost the totality of those packets which failed previously.

Withal, *FCSErrorCount* seems to have the less correlated behavior since it is based on received frames while the other two parameters depend on transmitted frames. In general, we can see that it keeps next to the ‘zero’ line and its values fluctuate between positive and negative.

5.3.5. Locating APs

Also connected to MAC counters, we can study particularities of the network and consider them with relation to the working environment.

During the time the system has been running, we have observed that every AP has 11.7 neighbors and 9.68 rogue neighbors on average. In particular, the AP seeing the largest number of neighbors is AP60[6] while AP64[7] is the one who has more rogue neighbors.

This data makes sense if we analyze where these APs are located. Both AP60 and AP64 belong to the third floor of EETAC building. In particular, AP64 provides coverage to ENTEL department (where the Wireless Network Group’s lab is located). AP60 is located in the third floor of the classrooms’ area. There is a wide view of all the campus from there, without obstacles blocking the direct line of sight between APs from different buildings.

Through MAC counters we can also find out which APs are the most aggrieved. We are aware that traffic characteristics vary every day and are not predictable. In spite of that, measures have been taken during a prolonged time, so statistical values provide interesting information, as it is shown in the tables attached in ANNEX B.

By observing them, we can affirm that there is a correlation between the prior configuration and the assignments we proposed since many APs repeat in the ‘ranking’ for all three counters studied.

[6] This AP is called “XSFEPSC40” according to UPCNET terminology

[7] This AP is called “XSFEPSC41” according to UPCNET terminology

In the case of *FailedCount* counters, we can see that most of the APs occupy similar positions with respect to the prior analysis. In general terms, we note that firsts positions belong to APs installed in lecture room buildings. Besides these, AP64 (the one giving coverage to ENTEL department) appears as the most problematic one (in terms of *FailedCount* value).
Referred to *MultipleRetryCount* and *FCSErrorCount* classification, it is noteworthy that all APs having the highest values are those installed at the library and the canteen. After them, next positions in the ranking belong to APs installed in classrooms in EETAC building.

To see where exactly in the campus the most relevant APs are installed, address to ANNEX C.

5.4. Beyond MAC counters

Besides all the parameters related to network performance, this process also deals with other costs, which do not belong to the guts but affect the performance. They are exposed within this section.

5.4.1. SNMP data

In the very first stage as well as during the time the system has been running, the extraction of SNMP parameters has been repeated several times a day.
On the one hand, we have executed *snmpwalk* commands. Through the analysis of traffic with Wireshark[8] (see Fig. 5. 9), we can say that every request to the controller takes around 21.7 seconds. During this time, there is traffic of 45,000 packets containing information for its later analysis.

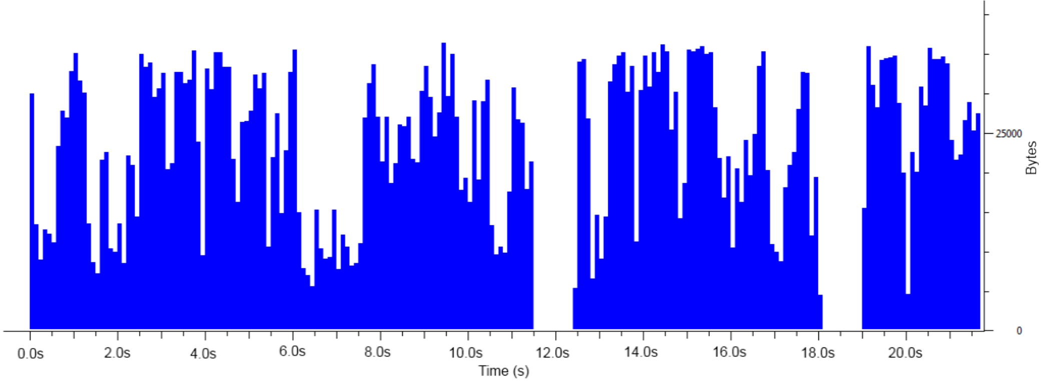


Fig. 5.9. Wireshark capture (1)

As a result, every time the script is executed there is a total amount of 4.72 MB of traffic generated. Considering it is run four times a day (before and after every peak of traffic), it represents a daily sum of 19 MB.

[8] http://www.wireshark.org/about.html

On the other hand, we have also executed *snmpset* commands intended to set two objects for each AP requiring a new channel (i.e. *ChannelAssignment* and *ChannelNumber*). The worst case, in which we would change the channel of all the managed APs, would imply 61,44 seconds and 40.7 KB of traffic (see Fig. 5. 10). Note that this time is higher than the *snmpget* one. This is so because we decided to pause the system after each *snmpset* command with the aim of avoiding a synchronized channel change in all of the APs, not overloading the controller with SNMP commands and to avoid changing the channel of all APs at the same time, which could potentially produce a flooding of signaling from stations looking for their APs.

Nevertheless, let us remember that changes are just applied to a reduced set of APs. During the test, a new assignment represented configuration changes for 20% of the APs on average, requiring only 12 seconds.

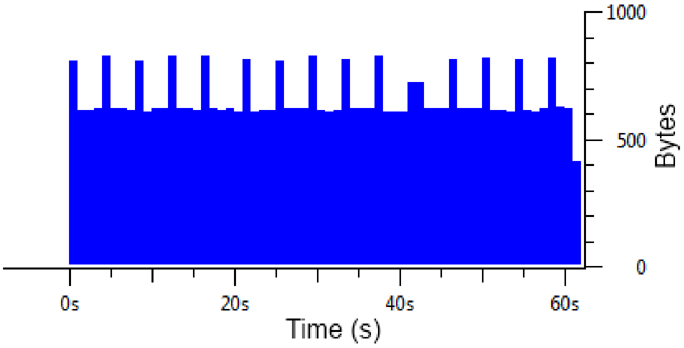


Fig. 5.10. Wireshark capture (2)

5.4.2. Effects observed by the user

Since changes in the network configuration were applied late during the night, when no users were using the WLAN, nobody was affected by the changes applied on the working scenario. But anyway, it is important to consider to what extent channel changes can affect users working on the wireless network of the campus.

Measurements carried out with this purpose establish that the disruption time a user can suffer after a sudden channel change by its current AP varies between 0.8 and 4.7 seconds, in the worst case. This difference is basically due to the Extended Service Set IDentifier (ESSID) to which the user associates (XSF-UPC or eduroam), as each one uses a different authentication method. We have also observed that the disconnection depends on the channel distance between the previous configuration and the new channel set.

As we see in both extremes of the disconnection time, we can conclude that the fact of changing the frequency in which an AP works is almost transparent for the final user, given that even the TCP session will not expire.

Seen that channel configuration is almost harmless to the user, we suggest carrying out these changes more frequently. This way, APs' working channel would change more dynamically according to the specific network conditions; and thus, the network would always work at its best performance.

5.5. Final considerations

SNMP information has been collected at those times matching the initial and the latest moments of each of the two utilization peaks existing in a regular day. This decision was taken before starting the stage devoted to analyze AP statistics, and we have relied on it until the end of the tests. However, we are aware that the traffic distribution varies every day, and it could affect MAC counters values to some extent.

Besides, we should also take into account that we are working in a real environment and there are factors out of the scope of the system. In general, failures attributed to the transmission and reception of packets can be assigned to several facts, such as interference, packet collision or low quality of a wireless link. Among them, our frequency management scheme acts directly on the two first issues, but does not have any impact on the performances issues caused by signal quality. That being said, the quality of the signal that an AP receives is affected by several factors, such as distance, noise or location. These are unpredictable and vary continuously so that, together with the varying traffic demands, it explains the variations suffered among the days the system has been analyzed.

Another factor that may affect our results is the operating mode used to run the system. It consists in setting the new channel assignment at night according to the traffic statistics of that day, and exploiting it the next working day. This procedure would be ideal provided that the general network characteristics remain in consecutive days. However, every day is different from a traffic point of view. Hence, some days we can find a channel distribution which is not the optimal for the day it is used; and as a result, there is still room for new improvements.

Otherwise, at the start of this chapter we have already discussed the decision of analyzing the *FailedCount*, *MultipleRetryCount* and *FCSErrorCount* MAC counters after the detection of a known bug in the Cisco WLAN Controller software release installed; this fact prevented us from calculating PER, which was the preferred performance metric. Although the problem acknowledged by Cisco just affected a given set of MAC counters, we cannot dismiss the possibility that other counters are also affected. Nonetheless, we consider that the results observed and the conclusions derived are reliable.

Finally, we would like to add the following: along the time this study has been accomplished, the wireless network where the test has been carried out has not suffered from any incidences due to the operation of the system we present within this work.

CHAPTER 6. CONCLUSIONS AND FUTURE WORK

This work comes to an end with the present chapter. It contains a final discussion including the evaluation of the results obtained, as well as those conclusions taken along the process. To conclude, some future lines of work are suggested so that enhancements in this communications area can continue.

6.1. Conclusions

The work presented within these lines contributes on getting the maximum performance of WLANs through the implementation of a channel management system, with the singularity of doing it in a real scenario.

In general, the process followed during these months can be divided into several stages. It started with an initial touchdown of Cisco APs in which technical characteristics, command-line interface and also available SNMP parameters were studied. Then it was time to do an exhaustive analysis over traffic generated by the users of the CBLs' wireless network as well as a research among SNMP parameters available in the software running at the central controller. Taking that information as a reference, we have been able to design a system which communicates with the central controller, periodically extract statistics about the performance of the WLAN and daily configure the network according to the channel distribution established by a known algorithm. Finally, the system has been run in CBL-UPC Barcelona Tech during several weeks, during which the results were studied in order to get quantifiable measurements.

With the aim of obtaining quantifiable measurements, we were supposed to study certain MAC counters intended to obtain PER, as the metric of the performance. However, the detection of a bug in the Cisco WLAN Controller software release installed, led us to the analysis of other MAC counters: *FailedCount*, *MultipleRetryCount* and *FCSErrrorCount*. The analysis of those counters has given satisfactory results, both in transmission and reception links.
Benefits obtained in reception link have been analyzed through *FCSErrorCount* counter. The analysis of this parameter has reported important decreases; we measured half the reception errors compared with Cisco's default configuration.
Through *MultipleRetryCount* and *FailedCount* we have been able to measure improvements related to transmission from the managed APs. *FailedCount* is the most critical parameter since an increase of its value means a failure in the transmission of a packet despite several attempts. In this case, changes applied in the frequency channel assignment make us experience a 75% of improvement on average. Nevertheless, *MultipleRetryCount* value has increased significantly, which is a consistent result with the improvement involving PER, as explained in CHAPTER 5.

The enhancements provided by our system are in part due to the algorithm chosen, capable of deciding the most appropriate channel assignment starting with a suitable text representation of the wireless medium, as seen from the managed APs. The importance of this mechanism relies on several facts: besides taking both co-channel and adjacent interference into account, it also uses all the available channels of the spectrum instead of the traditional non-overlapping three. Through these considerations, collisions and transmission errors are minimized, thus improving network capacity and user experience.

That being said, objectives proposed at the beginning of the work have been successfully achieved.

Besides, during the writing of the state of the art we realized that research done in real scenarios was limited and hence, a new challenge arose. That limited research is probably due to several facts, such as the ease and simplicity of testing a simulated environment, the special singularities (available SNMP data, users distribution, traffic pattern, …) every network owns, and especially the complexity of testing a campus-wide network during normal operation.

Even so, our system approach has been tested during near a month of typical traffic at CBL-UPC Barcelona Tech, obtaining favourable outcomes as a result. In addition, it is worth mentioning that during the time our channel management system has been running, there has not been any incidence in the wireless network attributable to it.

After the positive experience this work has brought, it will open the door to further studies which could continue extracting the best of WLANs in real environments; either continuing with the future work this closure suggests, or accepting new challenges, as the ones we propose in the section below.

The environmental impact involving wireless networks is minimal since the current legislation regarding the use of radioelectrical spectrum establishes that devices working in the 2.4GHz band must transmit at a low power level. Moreover, as of today there is no evidence of negative effects neither in the environment nor on people's health.

What is more, this work is intended to minimize interference and obtain the best network performance. In doing so, the improbable harmful effects of radiation from WLANs are further reduced.

Beyond academic achievements, all the process involving this Master Thesis has been certainly rewarding, and the best choice to finish this formation. It has allowed me to get familiar with IEEE 802.11, a technology deeply integrated in our daily life; and through the development of this work, taking part in getting its maximum performance.

There are many skills acquired or consolidated during this time: from the initial touchdown of Cisco APs and the search of interesting SNMP parameters for our purpose, to the skills acquired while implementing the system; every single point has brought something new.

6.2. Future work

Since the system presented is an initial approach, several lines of work have arisen during its development. All of them share the same aim: use frequency channel management to continue fighting interference and performance degradation in IEEE 802.11 WLANs. And thereby, achieve WLANs to deliver its maximum performance

The issue with the highest priority to be solved is the evaluation of PER, so that the upload and download error rate of the overall network could be used as the most reliable metric of the performance. This way, every AP performance will be measured as a ratio (%), relating packet error count to the total number of packets processed. This matter could be easily carried out once network managers update Cisco WLAN Controller software release running in the central controller.

Another matter closely related to this work is the evaluation of the system performance under a high number of concurrent users, so that APs (or a reduced set of them) work close to their maximal capacity. By means of the execution of this load test, we could learn how the network behaves under both normal and peak conditions and, in this way, evaluate whether any characteristic intensifies its results, as well as the reliability and scalability of the system.

Beyond these pending tasks concerning complementary issues to the study presented, other lines of work can also be followed. They are proposed in the following:

Another interesting factor noticed when evaluating the system performance was the number of APs affected by the daily channel reconfiguration. According to the analysis of data, twenty APs, on average, were assigned a new channel every day. Measurements carried out establish also that these changes are almost harmless to the user. Therefore, we propose a renewed system which could be applied more than once a day. What is more, it could be a great idea to adapt this channel management system to be more dynamic so that channel assignment was the optimal according to the precise needs of the network in every moment of a working day.

To go further in this way of getting the best out of a WLAN, we finally propose to carry out this study in a distributed network system. Contrary to centralized LANs, distributed ones are based on autonomous elements (APs) which are able to take decisions without the need of a central controller. Therefore, by adapting it to such architecture and test the improvements it provides, we could assure the enhancements reached in any enterprise WLAN scheme

REFERENCES

[1] E. García Villegas, Self-Optimization of Radio Resources on IEEE 802.11 Networks, July 2009.

[2] "http://eetac.upc.edu/ca/," [Online].

[3] "www.upc.edu," [Online].

[4] "http://www.entel.upc.edu/," [Online].

[5] "http://www.upcnet.es/," [Online].

[6] "Wi-Fi Alliance," [Online]. Available: http://www.wi-fi.org.

[7] F. L. F. R. E. V. F. Z. R. Nobel, "Planning and Designing 802.11 Wireless Technologies," *Cisco Press,* May 2012.

[8] K. A. E. B. L.B. Deek, "Channel Management for 802.11n Wireless Deployments," Santa Barbara, California, 2010.

[9] S. Yu, "IEEE 802.11 expanded support faster, higher-quality, simpler wireless LAN communications in more environments," 2012. [Online]. Available: http://standards.ieee.org/news/2012/802.11-2012.html.

[10] A. P. N. Prasad, WLAN Systems and Wireless IP for Next Generation Communications, 2002.

[11] "AIRESPACE-WIRELESS-MIB," 2010. [Online]. Available: http://www.oidview.com/mibs/14179/AIRESPACE-WIRELESS-MIB.html.

[12] G.Conradi, "Current status and Overview of the CAPWAP Protocol," [Online]. Available: http://www.cse.wustl.edu/~jain/cse574-10/ftp/capwap/index.html.

[13] W. M. G. Kbar, "Distributed Resources Management in Wireless LANs," in *The Second International Conference of Innovations in Information Technology*, 2005.

[14] Y. Berejano and S.-J. Han, "Fairness and Load Balancing in Wireless LANs Using Association Control," in *Proc. of the 10th international conference on Mobile Computing and Networking, MobiCom'04*, 2004.

[15] B. Alawieh, Y. Zhang, A. C. and M. H., "Improving Spatial Reuse in Multihop Wireless Networks. A Survey," *IEEE Community Surveys and Tutorials,* vol. 11, no. 3, 2009.

[16] S. Chieochan and E. D. J. Hossain, "Channel Assignment Schemes for Infrastructure-Based 802.11 WLANs: A Survey," *IEEE Communications Surveys & Tutorials,* vol. 12, no. 1, 1st quarter 2010, 2010.

[17] K. Zhou, L. Xie, Y. Chang and X. Tang, “Channel Assignment for WLAN by Considering Overlapping Channels in SINR Interference Model,” in *IEEE International Conference on Computing Networking and Communications*, 2012.

[18] E. Garcia, E. López-Aguilera, R. Vidal and J. Paradells, "Effect of adjacent-channel interference in IEEE 802.11 WLANs," in *2nd. Int. Conference of Cognitive Radio Oriented Wireless Networks and Communications, CrownCom'07*, 2007.

[19] Y. Cui, W. Li and X. Cheng, “Partially Overlapping Channel Assignment Based on "Node Orthogonality" for 802.11 Wireless Networks,” in *IEEE INFOCOM 2011*, 2011.

[20] F. Gamba, Wagen, Jean-Frédéric and D. Rossier, "Towards Adaptative WLAN Frequency Management Using Intelligent Agents," in *ADHOC-NOW'03*, Heidelberg, 2003.

[21] Y. Wang, L. Cuthbert and J. Bigham, “Intelligent RAdio Resource Management for IEEE 802.11 WLAN,” in *WCNC'04*, 2004.

[22] Y. Liu, W. Wu, B. Wang, T. He, S. Yi and Y. Xia, "Measurement-Based Channel Management in WLANs," in *WCNC'10*, 2010.

[23] M. Yu, H. Luo and K. K. Leung, “A Dynamic Radio Resource Management Technique for Multiple APs in WLANs,” *IEEE Transactions on Wireless Communications,* vol. 5, no. 7, 2006.

[24] V. Angelakis, S. Papadakis, V. A. Siris and A. Traganitis, "Adjacent Channel Interference in 802.11a is Harmful: Testbed Validation of a Simple Quantification Model," *IEEE Communications Magazine,* pp. 160-166, March 2011.

[25] Y. h. Kim, J. Jeong and C.-k. Kim, “Adjacent Channel Interference Aware Channel Assignment Scheme for WLANs,” in *ICUFN 2010*, 2010.

[26] Y. Liu, Y. Xiong, Y. Yang, P. Xu and Q. Zhang, "An experimental study on multi-channel multi-radio," in *Proc. IEEE GLOBECOM'05*, 2005.

[27] V. Angelakis, A. Traganitis and V. Siris, "Adjacent channel interference in a multi-radio wireless mesh node with 802.11a/g interfaces," in *Proc. IEEE INFOCOM'07*, 2007.

[28] E. Garcia Villegas, R. Vidal Ferré and J. Paradells, “Frequency assignments in IEEE 802.11 WLANs with efficient spectrum sharing,” *Wireless*

Communications and Mobils Computing, pp. 1125-1140, 2009.

[29] “Cisco Systems Inc.,” [Online]. Available: www.cisco.com.

[30] “Cisco Wireless LAN Controllers,” [Online]. Available: http://www.cisco.com/en/US/prod/collateral/wireless/ps6302/ps8322/ps6307/product_data_sheet0900aecd802570b0_ps6366_Products_Data_Sheet.html.

[31] “Airespace MIB,” [Online]. Available: http://www.oidview.com/mibs/14179/AIRESPACE-WIRELESS-MIB.html.

[32] “iReasoning MIB Browser,” [Online]. Available: http://ireasoning.com/mibbrowser.shtml.

[33] “Net-SNMP,” [Online]. Available: http://www.net-snmp.org/.

[34] W. Klotz, “Graph Coloring Algorithms,” 2000.

[35] R. H. A.B.Sediq, “Optimal Tradeoff between Efficiency and Jain's Fairness Index in Resource Allocation”.

[36] “IEEE802.11,” [Online]. Available: http://www.ieee802.org/11/.

ABRREVIATIONS AND ACRONYMS

ACI	Adjacent Channel Interference
ACK	Acknowledgement
AP	Access Point
ASN	Abstract Syntax Notation
BSS	Basic Service Set
BSSID	Basic Service Set Identifier
CAPWAP	Control And Provisioning of Wireless Access Points
CBL	Campus Baix Llobregat
CC	Central Controller
CIMNE	Centre Internacional de Mètodes Numèrics a l’Enginyeria
CSMA	Carrier Sense Multiple Access
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
DFS	Dynamic Frequency Selection
CTS	Clear to Send
DSATUR	Degree of Saturation
DSSS	Direct Sequence Spread Spectrum
DTLS	Datagram Transport Layer Security
EETAC	Escola d'Enginyeria de Telecomunicació i Aeroespacial de Castelldefels
ESAB	Escola Superior d’Agriculutra de Barcelona
ESS	Extended Service Set
ESSID	Extended Service Set IDentifier
ETSI	European Telecommunications Standard Institute
FCS	Frame Check Sequence
FHSS	Frequency Hopping Spread Spectrum
GUI	Graphical User Interface
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
IE	Internet Explorer
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
ISM	Industrial, Scientific and Medical band
ISO	International Organization for Standardization
LAN	Local Area Network
LAP	Lightweight Access Point
LWAPP	Light Weight Access Point Protocol
MAC	Media Access Control
MSDU	MAC Service Data Unit
MIB	Management Information Base
NMS	Network Management Systemantes

OID	Object IDentifier
PDU	Protocol Data Unit
PER	Packet Error Ratio
PHY	Physical (layer)
QoE	Quality of Experience
RDIT	Recerca, Desenvolupament i Innovació Tecnològica
RFC	Request For Comments
RTS	Request To Send
SINR	Signal to Interference Noise Ratio
SNMP	Simple Network Management Protocol
SNR	Signal to Noise Ratio
SSH	Secure SHell
SSID	Service Set IDentifier
STA	Station
UDP	User Datagram Protocol
UPC	Universitat Politècnica de Catalunya
USA	United States of America
WLAN	Wireless Local Area Network
WLC	Wireless LAN Controller
WG	Working Group

UPC eetac Escola d'Enginyeria de Telecomunicació i Aeroespacial de Castelldefels

UNIVERSITAT POLITÈCNICA DE CATALUNYA

ANNEXES

TITLE: Frequency management in a campus-wide Wi-Fi deployment

MASTER DEGREE: Master in Science in Telecommunication Engineering & Management

AUTHOR: Ester Mengual Pérez

DIRECTOR: Eduard Garcia Villegas

DATE: February 18, 2013

ANNEX A EXAMPLE SCENARIO

The information presented in this annex is intended to complement the process already explained in chapter 4.2 providing files content and results to the simulated scenario used for this purpose.

A.1 Example scenario

The example scenario created is composed of six APs (ID:0 to ID:5) and three rogue APs (ID:6 to ID:8) connected as follows:

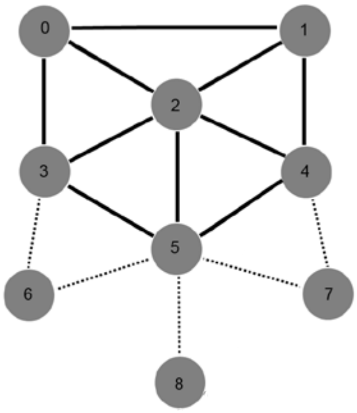


Fig. A.1. Simulated scenario

A.2 Getting SNMP information

SNMP information is taken through a series of commands based on the following one:

```
snmpwalk -v2c -On -c **password** **controllerIP** 1.3.6.1.4.1.14179.2.2.1.1
```

This set of requests result in a file containing the information provided below:

```
//Information regarding frequency channel
.1.3.6.1.4.1.14179.2.2.2.1.4.0.15.36.209.90.32.0 = INTEGER: 11
.1.3.6.1.4.1.14179.2.2.2.1.4.0.15.36.209.90.96.0 = INTEGER: 1
.1.3.6.1.4.1.14179.2.2.2.1.4.0.15.36.212.14.64.0 = INTEGER: 1
.1.3.6.1.4.1.14179.2.2.2.1.4.0.15.36.214.151.32.0 = INTEGER: 1
.1.3.6.1.4.1.14179.2.2.2.1.4.0.15.36.214.156.128.0 = INTEGER: 6
.1.3.6.1.4.1.14179.2.2.2.1.4.0.15.36.214.162.224.0 = INTEGER: 1

//Information regarding RX utilization
.1.3.6.1.4.1.14179.2.2.13.1.1.0.15.36.209.90.32.0 = INTEGER: 0
.1.3.6.1.4.1.14179.2.2.13.1.1.0.15.36.209.90.96.0 = INTEGER: 0
```

```
.1.3.6.1.4.1.14179.2.2.13.1.1.0.15.36.212.14.64.0 = INTEGER: 0
.1.3.6.1.4.1.14179.2.2.13.1.1.0.15.36.214.151.32.0 = INTEGER: 0
.1.3.6.1.4.1.14179.2.2.13.1.1.0.15.36.214.156.128.0 = INTEGER: 0
.1.3.6.1.4.1.14179.2.2.13.1.1.0.15.36.214.162.224.0 = INTEGER: 0

//Information regarding TX utilization
.1.3.6.1.4.1.14179.2.2.13.1.2.0.15.36.209.90.32.0 = INTEGER: 0
.1.3.6.1.4.1.14179.2.2.13.1.2.0.15.36.209.90.96.0 = INTEGER: 0
.1.3.6.1.4.1.14179.2.2.13.1.2.0.15.36.212.14.64.0 = INTEGER: 0
.1.3.6.1.4.1.14179.2.2.13.1.2.0.15.36.214.151.32.0 = INTEGER: 1
.1.3.6.1.4.1.14179.2.2.13.1.2.0.15.36.214.156.128.0 = INTEGER: 0
.1.3.6.1.4.1.14179.2.2.13.1.2.0.15.36.214.162.224.0 = INTEGER: 0

//Information regarding channel utilization
.1.3.6.1.4.1.14179.2.2.13.1.3.0.15.36.209.90.32.0 = INTEGER: 15
.1.3.6.1.4.1.14179.2.2.13.1.3.0.15.36.209.90.96.0 = INTEGER: 2
.1.3.6.1.4.1.14179.2.2.13.1.3.0.15.36.212.14.64.0 = INTEGER: 14
.1.3.6.1.4.1.14179.2.2.13.1.3.0.15.36.214.151.32.0 = INTEGER: 16
.1.3.6.1.4.1.14179.2.2.13.1.3.0.15.36.214.156.128.0 = INTEGER: 9
.1.3.6.1.4.1.14179.2.2.13.1.3.0.15.36.214.162.224.0 = INTEGER: 0

//Information regarding AP neighbors
.1.3.6.1.4.1.14179.2.2.17.1.1.0.15.36.209.90.32.0.0.15.36.209.90.96 = Hex-STRING: 00 0F 24 D1 5A 60
.1.3.6.1.4.1.14179.2.2.17.1.1.0.15.36.209.90.32.0.0.15.36.212.14.64 = Hex-STRING: 00 0F 24 D4 0E 40
.1.3.6.1.4.1.14179.2.2.17.1.1.0.15.36.209.90.32.0.0.15.36.214.151.32 = Hex-STRING: 00 0F 24 D6 97 20
.1.3.6.1.4.1.14179.2.2.17.1.1.0.15.36.209.90.96.0.0.15.36.209.90.32 = Hex-STRING: 00 0F 24 D1 5A 20
.1.3.6.1.4.1.14179.2.2.17.1.1.0.15.36.209.90.96.0.0.15.36.212.14.64 = Hex-STRING: 00 0F 24 D4 0E 40
.1.3.6.1.4.1.14179.2.2.17.1.1.0.15.36.209.90.96.0.0.15.36.214.156.128 = Hex-STRING: 00 0F 24 D6 9C 80
.1.3.6.1.4.1.14179.2.2.17.1.1.0.15.36.212.14.64.0.0.15.36.209.90.32 = Hex-STRING: 00 0F 24 D1 5A 20
.1.3.6.1.4.1.14179.2.2.17.1.1.0.15.36.212.14.64.0.0.15.36.209.90.96 = Hex-STRING: 00 0F 24 D1 5A 60
.1.3.6.1.4.1.14179.2.2.17.1.1.0.15.36.212.14.64.0.0.15.36.214.151.32 = Hex-STRING: 00 0F 24 D6 97 20
.1.3.6.1.4.1.14179.2.2.17.1.1.0.15.36.212.14.64.0.0.15.36.214.156.128 = Hex-STRING: 00 0F 24 D6 9C 80
.1.3.6.1.4.1.14179.2.2.17.1.1.0.15.36.214.151.32.0.0.15.36.209.90.32 = Hex-STRING: 00 0F 24 D1 5A 20
.1.3.6.1.4.1.14179.2.2.17.1.1.0.15.36.214.151.32.0.0.15.36.212.14.64 = Hex-STRING: 00 0F 24 D4 0E 40
.1.3.6.1.4.1.14179.2.2.17.1.1.0.15.36.214.151.32.0.0.15.36.214.162.224 = Hex-STRING: 00 0F 24 D6 A2 E0
.1.3.6.1.4.1.14179.2.2.17.1.1.0.15.36.214.156.128.0.0.15.36.209.90.96 = Hex-STRING: 00 0F 24 D1 5A 60
.1.3.6.1.4.1.14179.2.2.17.1.1.0.15.36.214.156.128.0.0.15.36.212.14.64 = Hex-STRING: 00 0F 24 D4 0E 40
.1.3.6.1.4.1.14179.2.2.17.1.1.0.15.36.214.156.128.0.0.15.36.214.162.224 = Hex-STRING: 00 0F 24 D6 A2 E0
.1.3.6.1.4.1.14179.2.2.17.1.1.0.15.36.214.162.224.0.0.15.36.214.151.32 = Hex-STRING: 00 0F 24 D4 0E 40
.1.3.6.1.4.1.14179.2.2.17.1.1.0.15.36.214.162.224.0.0.15.36.212.14.64 = Hex-STRING: 00 0F 24 D6 97 20
.1.3.6.1.4.1.14179.2.2.17.1.1.0.15.36.214.162.224.0.0.15.36.214.156.128 = Hex-STRING: 00 0F 24 D6 9C 80

//Information regarding Neighbor-AP RSSI
.1.3.6.1.4.1.14179.2.2.17.1.3.0.15.36.209.90.32.0.0.15.36.209.90.96 = INTEGER: -75
.1.3.6.1.4.1.14179.2.2.17.1.3.0.15.36.209.90.32.0.0.15.36.212.14.64 = INTEGER: -75
.1.3.6.1.4.1.14179.2.2.17.1.3.0.15.36.209.90.32.0.0.15.36.214.151.32 = INTEGER: -75
.1.3.6.1.4.1.14179.2.2.17.1.3.0.15.36.209.90.96.0.0.15.36.209.90.32 = INTEGER: -75
.1.3.6.1.4.1.14179.2.2.17.1.3.0.15.36.209.90.96.0.0.15.36.212.14.64 = INTEGER: -75
.1.3.6.1.4.1.14179.2.2.17.1.3.0.15.36.209.90.96.0.0.15.36.214.156.128 = INTEGER: -75
.1.3.6.1.4.1.14179.2.2.17.1.3.0.15.36.212.14.64.0.0.15.36.209.90.32 = INTEGER: -75
.1.3.6.1.4.1.14179.2.2.17.1.3.0.15.36.212.14.64.0.0.15.36.209.90.96 = INTEGER: -75
.1.3.6.1.4.1.14179.2.2.17.1.3.0.15.36.212.14.64.0.0.15.36.214.151.32 = INTEGER: -75
.1.3.6.1.4.1.14179.2.2.17.1.3.0.15.36.212.14.64.0.0.15.36.214.156.128 = INTEGER: -75
.1.3.6.1.4.1.14179.2.2.17.1.3.0.15.36.214.151.32.0.0.15.36.209.90.32 = INTEGER: -75
.1.3.6.1.4.1.14179.2.2.17.1.3.0.15.36.214.151.32.0.0.15.36.212.14.64 = INTEGER: -75
.1.3.6.1.4.1.14179.2.2.17.1.3.0.15.36.214.151.32.0.0.15.36.214.162.224 = INTEGER: -75
.1.3.6.1.4.1.14179.2.2.17.1.3.0.15.36.214.156.128.0.0.15.36.209.90.96 = INTEGER: -75
.1.3.6.1.4.1.14179.2.2.17.1.3.0.15.36.214.156.128.0.0.15.36.212.14.64 = INTEGER: -75
.1.3.6.1.4.1.14179.2.2.17.1.3.0.15.36.214.156.128.0.0.15.36.214.162.224 = INTEGER: -75
```

```
.1.3.6.1.4.1.14179.2.2.17.1.3.0.15.36.214.162.224.0.0.15.36.214.151.32 = INTEGER: -75
.1.3.6.1.4.1.14179.2.2.17.1.3.0.15.36.214.162.224.0.0.15.36.212.14.64 = INTEGER: -75
.1.3.6.1.4.1.14179.2.2.17.1.3.0.15.36.214.162.224.0.0.15.36.214.156.128 = INTEGER: -75

//Information regarding MAC counters
.1.3.6.1.4.1.14179.2.2.6.1.1.0.15.36.209.90.32.0 = Counter32: 432209
.1.3.6.1.4.1.14179.2.2.6.1.1.0.15.36.209.90.96.0 = Counter32: 148292
.1.3.6.1.4.1.14179.2.2.6.1.1.0.15.36.212.14.64.0 = Counter32: 2981089
.1.3.6.1.4.1.14179.2.2.6.1.1.0.15.36.214.151.32.0 = Counter32: 858793
.1.3.6.1.4.1.14179.2.2.6.1.1.0.15.36.214.156.128.0 = Counter32: 2981089
.1.3.6.1.4.1.14179.2.2.6.1.1.0.15.36.214.162.224.0 = Counter32: 432209
.1.3.6.1.4.1.14179.2.2.6.1.3.0.15.36.209.90.32.0 = Counter32: 577881
.1.3.6.1.4.1.14179.2.2.6.1.3.0.15.36.209.90.96.0 = Counter32: 224466
.1.3.6.1.4.1.14179.2.2.6.1.3.0.15.36.212.14.64.0 = Counter32: 2297323
.1.3.6.1.4.1.14179.2.2.6.1.3.0.15.36.214.151.32.0 = Counter32: 1078132
.1.3.6.1.4.1.14179.2.2.6.1.3.0.15.36.214.156.128.0 = Counter32: 2297323
.1.3.6.1.4.1.14179.2.2.6.1.3.0.15.36.214.162.224.0 = Counter32: 577881
.1.3.6.1.4.1.14179.2.2.6.1.9.0.15.36.209.90.32.0 = Counter32: 0
.1.3.6.1.4.1.14179.2.2.6.1.9.0.15.36.209.90.96.0 = Counter32: 0
.1.3.6.1.4.1.14179.2.2.6.1.9.0.15.36.212.14.64.0 = Counter32: 0
.1.3.6.1.4.1.14179.2.2.6.1.9.0.15.36.214.151.32.0 = Counter32: 0
.1.3.6.1.4.1.14179.2.2.6.1.9.0.15.36.214.156.128.0 = Counter32: 0
.1.3.6.1.4.1.14179.2.2.6.1.9.0.15.36.214.162.224.0 = Counter32: 0
.1.3.6.1.4.1.14179.2.2.6.1.11.0.15.36.209.90.32.0 = Counter32: 5222330
.1.3.6.1.4.1.14179.2.2.6.1.11.0.15.36.209.90.96.0 = Counter32: 6495263
.1.3.6.1.4.1.14179.2.2.6.1.11.0.15.36.212.14.64.0 = Counter32: 12104136
.1.3.6.1.4.1.14179.2.2.6.1.11.0.15.36.214.151.32.0 = Counter32: 19296565
.1.3.6.1.4.1.14179.2.2.6.1.11.0.15.36.214.156.128.0 = Counter32: 12104136
.1.3.6.1.4.1.14179.2.2.6.1.11.0.15.36.214.162.224.0= Counter32: 5222330

// Information regarding MAC address of the AP that detected the rogue.
.1.3.6.1.4.1.14179.2.1.8.1.1.0.14.142.122.44.198.0.15.36.214.151.32.0 = Hex-STRING: 00 0F 24 D6 97 20
.1.3.6.1.4.1.14179.2.1.8.1.1.0.14.142.122.44.198.0.15.36.214.162.224.0 = Hex-STRING: 00 0F 24 D6 A2 E0
.1.3.6.1.4.1.14179.2.1.8.1.1.0.3.47.32.153.130.0.15.36.214.162.224.0 = Hex-STRING: 00 0F 24 D6 A2 E0
.1.3.6.1.4.1.14179.2.1.8.1.1.0.15.102.77.74.40.0.15.36.214.162.224.0 = Hex-STRING: 00 0F 24 D6 A2 E0
.1.3.6.1.4.1.14179.2.1.8.1.1.0.15.102.77.74.40.0.15.36.214.156.128.0 = Hex-STRING: 00 0F 24 D6 9C 80

// Information regarding MAC address of the rogue APs
.1.3.6.1.4.1.14179.2.1.7.1.1.0.14.142.122.44.198 = Hex-STRING: 00 01 38 6D DE EE
.1.3.6.1.4.1.14179.2.1.7.1.1.0.3.47.32.153.130 = Hex-STRING: 00 01 38 D7 1E B1
.1.3.6.1.4.1.14179.2.1.7.1.1.0.15.102.77.74.40 = Hex-STRING: 00 01 38 DD CE B3

//Information regarding rogue channel
.1.3.6.1.4.1.14179.2.1.7.1.26.0.14.142.122.44.198 = INTEGER: 11
.1.3.6.1.4.1.14179.2.1.7.1.26.0.3.47.32.153.130 = INTEGER: 6
.1.3.6.1.4.1.14179.2.1.7.1.26.0.15.102.77.74.40 = INTEGER: 4

// Information regarding Rogue-AP RSSI
.1.3.6.1.4.1.14179.2.1.8.1.7.0.14.142.122.44.198.0.15.36.214.151.32.0 = INTEGER: -80
.1.3.6.1.4.1.14179.2.1.8.1.7.0.14.142.122.44.198.0.15.36.214.162.224.0 = INTEGER: -80
.1.3.6.1.4.1.14179.2.1.8.1.7.0.3.47.32.153.130.0.15.36.214.162.224.0 = INTEGER: -80
.1.3.6.1.4.1.14179.2.1.8.1.7.0.15.102.77.74.40.0.15.36.214.162.224.0 = INTEGER: -80
.1.3.6.1.4.1.14179.2.1.8.1.7.0.15.102.77.74.40.0.15.36.214.156.128.0 = INTEGER: -80
```

Among the total set of SNMP information collected from the getting commands, there is information about APs characteristics as well as about the rogue APs detected.

As it can be observed, when an AP is managed by the central controller, we can find plenty of information, such as its MAC address, frequency channel, RX, TX and channel utilization, MAC counters, AP neighbors and the RSSI with respect its neighbors. However, this is not what happens if we study rogue APs. In this case, the controller is just able to obtain information regarding its MAC address, frequency channel and the RSSI between the managed and the rogue AP.

A.3 SNMP data analysis

After processing the functions in charge of extracting useful SNMP information, the data extracted from the input file of this specific scenario (Fig. A. 1) will be organized as follows:

```
[0]OID:0.15.36.209.90.32.0 MAC:00 0F 24 D1 5A 20 .Channel:11. Load: 0,0,15.
MAC:00 0F 24 D1 5A 60 RSSI:-75 Times: 1 ID: 1
MAC:00 0F 24 D4 0E 40 RSSI:-75 Times: 1 ID: 2
MAC:00 0F 24 D6 97 20 RSSI:-75 Times: 1 ID: 3
[1]OID:0.15.36.209.90.96.0 MAC:00 0F 24 D1 5A 60 .Channel:1. Load: 0,0,2.
MAC:00 0F 24 D1 5A 20 RSSI:-75 Times: 1 ID: 0
MAC:00 0F 24 D4 0E 40 RSSI:-75 Times: 1 ID: 2
MAC:00 0F 24 D6 9C 80 RSSI:-75 Times: 1 ID: 4
[2]OID:0.15.36.212.14.64.0 MAC:00 0F 24 D4 0E 40. Channel:1. Load: 0,0,14.
MAC:00 0F 24 D1 5A 20 RSSI:-75 Times: 1 ID: 0
MAC:00 0F 24 D1 5A 60 RSSI:-75 Times: 1 ID: 1
MAC:00 0F 24 D6 97 20 RSSI:-75 Times: 1 ID: 3
MAC:00 0F 24 D6 9C 80 RSSI:-75 Times: 1 ID: 4
[3]OID:0.15.36.214.151.32.0 MAC:00 0F 24 D6 97 20. Channel:1. Load: 0,1,16.
MAC:00 0F 24 D1 5A 20 RSSI:-75 Times: 1 ID: 0
MAC:00 0F 24 D4 0E 40 RSSI:-75 Times: 1 ID: 2
MAC:00 0F 24 D6 A2 E0 RSSI:-75 Times: 1 ID: 5
MAC:00 01 38 6D DE EE RSSI:-80 Times: 1 ID: 6
[4]OID:0.15.36.214.156.128.0 MAC:00 0F 24 D6 9C 80. Channel:6. Load: 0,0,9.
MAC:00 0F 24 D1 5A 60 RSSI:-75 Times: 1 ID: 1
MAC:00 0F 24 D4 0E 40 RSSI:-75 Times: 1 ID: 2
MAC:00 0F 24 D6 A2 E0 RSSI:-75 Times: 1 ID: 5
MAC:00 01 38 DD CE B3 RSSI:-80 Times: 1 ID: 8
[5]OID:0.15.36.214.162.224.0 MAC:00 0F 24 D6 A2 E0. Channel:1. Load: 0,0,0.
MAC:00 0F 24 D4 0E 40 RSSI:-75 Times: 1 ID: 2
MAC:00 0F 24 D6 97 20 RSSI:-75 Times: 1 ID: 3
MAC:00 0F 24 D6 9C 80 RSSI:-75 Times: 1 ID: 4
MAC:00 01 38 6D DE EE RSSI:-80 Times: 1 ID: 6
MAC:00 01 38 D7 1E B1 RSSI:-80 Times: 1 ID: 7
MAC:00 01 38 DD CE B3 RSSI:-80 Times: 1 ID: 8
[6]OID:0.14.142.122.44.198 MAC:00 01 38 6D DE EE. Channel:11. Load: 0,0,0.
[7]OID:0.3.47.32.153.130 MAC:00 01 38 D7 1E B1. Channel:6. Load: 0,0,0.
[8]OID:0.15.102.77.74.40 MAC:00 01 38 DD CE B3. Channel:4. Load: 0,0,0.
```

Once all the SNMP information has been examined, it still remains output files to be created.

A.4 DSATUR input and output

In order to make the decision about the new channel occupied by every AP, Dsatur needs an input file containing the information in a predefined format.

The text representation of the graph must include information such as the represented table below. It is obtained through requests to the controller and includes *n* (number of nodes in the graph), *u* (utilization of each AP), *f* (the original frequency channel assigned to each AP), and *e* (representation of interference relationships between APs in terms of RSSI). A 0.0 in an interference *e* line position denotes that information is unknown).

```
n 9
u 0 0 0 1 0 0 0 0 0
f 11 1 1 1 6 1 11 6 4
e 0 1 -75.0 -75.0
e 0 2 -75.0 -75.0
e 0 3 -75.0 -75.0
e 1 2 -75.0 -75.0
e 1 4 -75.0 -75.0
e 2 3 -75.0 -75.0
e 2 4 -75.0 -75.0
e 3 5 -75.0 -75.0
e 3 6 0.0 -80.0
e 4 5 -75.0 -75.0
e 4 8 0.0 -80.0
e 5 2 0.0 -75.0
e 5 6 0.0 -80.0
e 5 7 0.0 -80.0
e 5 8 0.0 -80.0
```

After running the algorithm we get such an output as the given below. Among some other computational information, we also obtain the new channel assignment for every node of the entire network.

```
9 nodes in the graph
cost assignacio inicial: 164.147583
cost TOTAL: 214.227173
Millora goodput 30.508880%
8 colors usats
_____________________________________________________________
| Node  |  0 |  1 |  2 |  3 |  4 |  5 |  6 |  7 |  8 |
-------------------------------------------------------------
| Canal | 13 |  2 |  1 |  7 |  8 | 11 | 11 |  6 |  4 |

CPU time = 0.004000 sec.
```

A.5 Setting SNMP parameters

In the same way that a specific command allows the capture of SNMP packets, for the final part of the process we need to change some of them. The following request shows how snmpset can be accomplished:

```
snmpset -v2c -On -c **password** **controllerIP** **specificOID** i 2
snmpset -v2c -On -c **password** **controllerIP** **specificOID** i 6
```

ANNEX B RESULTS EVALUATION

The tables and images presented in this annex are intended to provide extra information about the analysis of results explained in Chapter 5. In particular, there are tables which provide details about the network configuration, as well as some images which are easier to analyze in a larger version.

B.1 Frequency channel assignment

According to the process followed, Dsatur has daily decided the most appropriated channel distribution in order to obtain the better performance of the network.
This daily channel distribution is presented in the following table:

Table B.1. Daily channel assignment

ID	PRE	d1	d2	d3	d4	d5	d6	d7	d8	d9	d10	d11	d12
0	**11**	11	3	3	3	3	3	3	3	3	3	3	3
1	**1**	9	9	9	9	9	9	9	9	1	6	9	9
2	**1**	2	2	2	2	2	2	2	2	2	2	2	1
3	**1**	13	13	13	1	2	1	3	3	13	3	3	1
4	**1**	1	13	13	13	13	13	13	13	13	13	13	13
5	**1**	1	1	7	1	10	10	10	10	10	13	13	13
6	**11**	12	12	12	12	12	12	12	3	3	13	13	13
7	**1**	1	1	6	8	8	8	8	8	8	8	8	8
8	**1**	5	5	1	4	4	1	3	3	3	3	3	1
9	**1**	9	9	9	9	9	9	9	9	9	9	9	9
10	**11**	12	12	12	12	12	12	12	13	13	13	13	13
11	**1**	2	2	1	1	1	1	1	1	1	1	1	1
12	**1**	6	9	8	9	9	13	13	13	13	13	5	9
13	**11**	13	13	13	13	13	13	13	13	13	13	13	13
14	**1**	2	2	2	2	2	11	4	4	4	5	8	8
15	**11**	11	1	13	13	13	13	13	13	1	1	1	1
16	**11**	6	6	6	6	6	13	13	1	8	13	13	13
17	**1**	10	10	10	10	10	10	10	9	9	9	9	9
18	**6**	4	4	4	4	4	4	4	4	4	4	4	1
19	**1**	13	13	13	13	13	13	13	13	13	13	13	13
20	**1**	1	1	2	9	1	6	13	1	1	8	8	1
21	**1**	1	9	6	5	9	1	1	9	5	13	9	5
22	**11**	10	1	2	1	1	2	8	8	8	1	13	13
23	**6**	13	13	13	13	13	13	13	13	13	2	13	5
24	**1**	1	1	1	10	10	10	10	10	10	10	10	9
25	**6**	6	6	6	6	6	6	6	6	6	6	6	6
26	**11**	11	13	13	13	2	11	11	11	12	13	13	11
27	**11**	11	11	11	9	9	9	9	9	9	9	9	9
28	**1**	11	13	13	13	13	13	13	13	13	13	13	13

ID	PRE	d1	d2	d3	d4	d5	d6	d7	d8	d9	d10	d11	d12
29	**11**	11	10	10	10	10	10	10	10	10	10	10	10
30	**1**	1	1	1	1	1	10	13	13	1	1	1	1
31	**1**	7	7	7	7	8	8	8	8	8	8	8	8
32	**1**	1	9	9	9	9	9	9	9	9	9	9	9
33	**1**	1	13	13	13	13	13	13	13	13	13	13	13
34	**1**	1	1	2	2	2	2	2	2	2	2	2	2
35	**11**	13	13	13	13	13	13	13	13	13	13	13	13
36	**1**	2	2	2	2	2	2	2	2	2	2	2	2
37	**11**	11	11	11	11	1	3	3	3	3	3	3	3
38	**6**	6	6	6	6	6	6	6	6	6	6	6	6
39	**1**	1	1	1	1	1	1	7	5	10	6	1	2
40	**1**	1	1	1	1	13	13	1	13	1	13	13	13
41	**11**	5	5	5	5	5	5	1	1	5	1	4	4
42	**1**	13	13	13	13	13	13	13	13	13	13	13	13
43	**1**	12	13	8	13	12	12	12	12	7	12	9	12
44	**6**	6	7	1	4	5	1	1	7	9	8	8	1
45	**6**	4	3	5	5	5	2	13	1	2	2	2	5
46	**1**	13	13	13	13	13	13	13	13	13	13	13	13
47	**6**	13	13	13	13	13	13	13	13	13	13	13	9
48	**11**	5	13	13	13	13	13	5	1	13	13	13	13
49	**11**	9	9	9	9	9	9	9	9	9	9	9	13
50	**11**	11	11	1	3	3	3	3	3	3	3	3	3
51	**11**	6	9	9	9	9	9	9	9	9	9	9	9
52	**6**	6	6	6	6	6	6	6	6	6	6	6	6
53	**6**	6	6	6	6	7	7	7	13	13	13	13	13
54	**1**	1	1	1	1	1	1	1	1	1	1	1	1
55	**11**	13	13	13	13	13	13	13	13	13	13	13	13
56	**1**	1	2	2	2	5	5	5	5	5	5	5	5
57	**1**	1	1	7	7	7	7	7	1	7	7	7	7
58	**1**	1	1	1	1	1	1	1	1	1	1	1	1
59	**6**	6	6	6	6	13	13	13	13	13	13	13	13
60	**6**	6	6	6	6	6	6	6	6	6	6	5	5
61	**1**	1	1	1	1	1	1	1	1	1	1	1	1
62	**1**	1	1	1	1	1	1	1	1	1	1	1	1
63	**11**	1	1	1	1	1	1	1	1	13	13	13	13
64	**6**	1	6	13	13	13	13	13	13	13	13	13	13
65	**1**	6	1	1	1	1	1	1	1	1	1	1	1
66	**11**	1	2	2	2	2	2	2	2	2	2	2	2
67	**11**	11	10	10	8	13	13	6	6	6	3	1	1
68	**11**	10	11	11	11	11	11	11	10	10	10	10	10
69	**6**	11	10	13	13	13	13	13	13	13	13	10	10
70	**1**	6	1	13	5	5	13	9	9	9	9	6	5
71	**6**	1	13	13	1	4	1	1	1	1	1	1	1
72	**6**	6	9	9	9	9	9	9	9	9	9	13	13
73	**6**	7	6	4	4	13	13	1	10	2	2	1	2
74	**1**	6	4	4	13	13	13	13	13	13	13	9	9
75	**6**	1	13	6	13	1	1	1	1	13	13	13	13
76	**11**	13	10	1	1	1	2	2	2	2	2	2	2
77	**6**	10	1	1	1	1	1	1	1	1	1	1	1

ID	PRE	d1	d2	d3	d4	d5	d6	d7	d8	d9	d10	d11	d12
78	**1**	1	1	1	1	1	1	1	1	1	1	1	1
79	**11**	1	9	9	9	9	5	5	13	13	13	13	13
80	**1**	2	9	9	9	9	1	10	1	10	10	12	10
81	**1**	1	1	13	13	13	13	13	13	13	13	13	13
82	**11**	1	13	13	13	11	12	12	13	13	13	13	13
83	**1**	13	1	1	9	9	9	10	13	11	13	13	13
84	**1**	1	11	11	13	13	13	13	13	13	13	13	13
85	**1**	1	1	1	1	1	1	1	1	1	1	1	1
86	**11**	1	11	11	11	11	11	11	11	11	11	11	11
87	**1**	11	1	1	1	1	4	4	4	4	4	4	4
88	**11**	1	11	11	7	7	7	7	7	7	7	7	7
89	**6**	11	1	1	1	1	13	13	13	13	13	13	13
90	**1**	6	1	1	2	2	2	2	2	2	2	2	2
91	**1**	1	1	1	1	1	1	1	1	1	1	13	13
92	**6**	1	7	7	7	7	7	7	7	7	7	7	7
93	**1**	7	1	1	1	1	1	1	1	1	1	1	1
94	**1**	1	1	1	1	1	1	1	1	2	2	2	1
95	**1**	1	1	1	1	1	1	8	8	8	5	13	13
96	**11**	1	13	13	13	13	13	13	13	13	13	13	13
97	**6**	11	6	13	13	13	13	13	13	13	13	13	13
98	**6**	6	5	5	5	5	5	5	5	5	5	5	5
99	**6**	6	6	6	6	6	6	13	13	13	13	13	13

From the information extracted in the table above (Table B. 1), we have obtained the graph shown in the next site (Fig. B. 1).

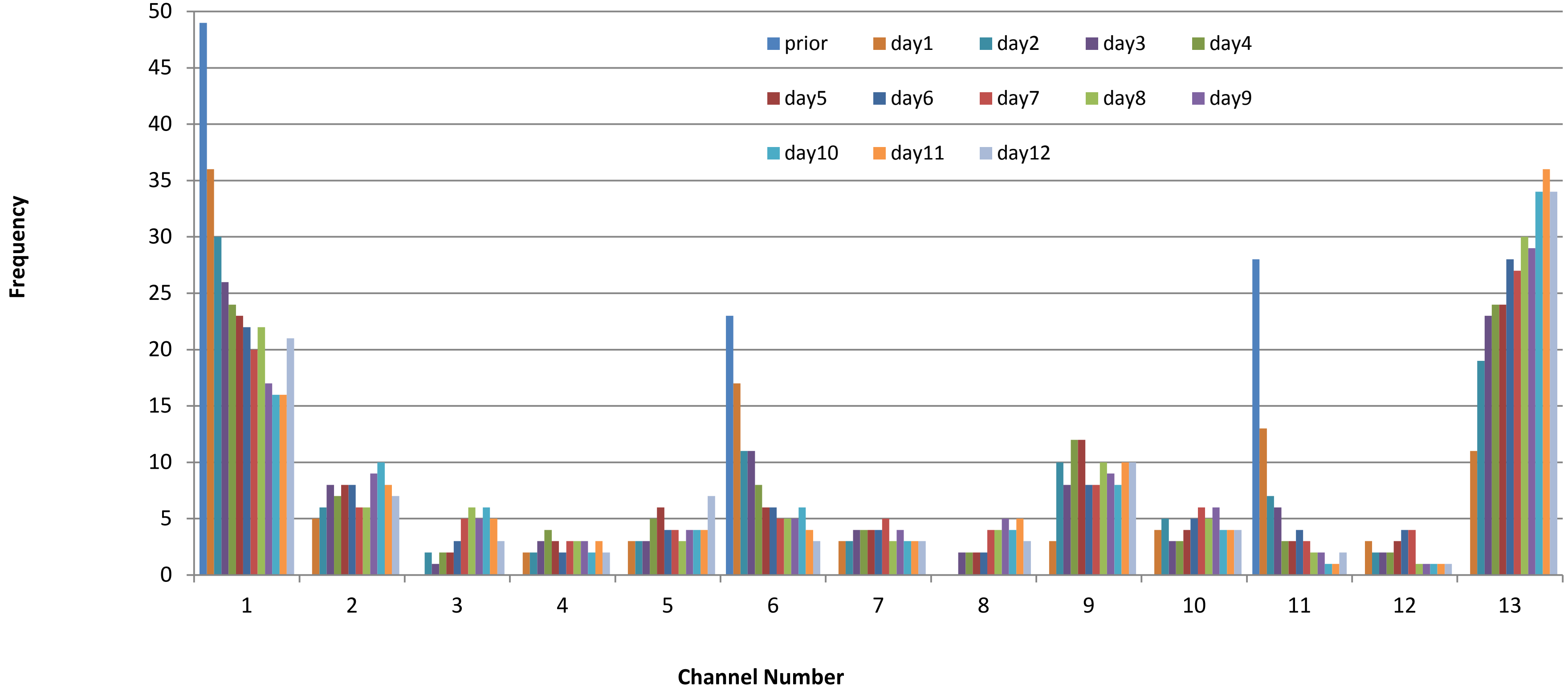


Fig. B.1. Channel assignment along time

B.2 MAC counters

Graphs depicted in Chapter 5 have been elaborated from those data representing the ten higher values of each MAC counter under study. Thus, it would be interesting to find out whether those APs keep being the same after changes or not.

This section shows a ranking of the counters (*FailedCount*, *MultipleRetryCount* and *FCSErrorCount*) both before and after running this system. So, information referred to them is respectively shown in Table B. 2, Table B. 3 and Table B. 4 .

Table B.2. IDs of the highest *FailedCount* values

Default Mode	Approach tested
96	64
50	96
55	50
0	86
66	0
7	1
9	66
62	10
98	7
87	99

Table B.3. IDs of the highest *MultipleRetryCount* values

Default Mode	Approach tested
15	23
99	10
23	2
16	99
10	16
24	14
11	20
62	21
85	12
19	7

Table B.4. IDs of the highest *FCSErrorCount* values

Default Mode	Approach tested
22	12
15	19
19	15
12	16
4	21
20	4
11	11
55	22
17	13
21	93

ANNEX C LOCATION OF APS

This annex contains drawings of the campus buildings where most of the relevant APs are located. The aim of this annex is complementing the information given in Chapter 5, where those APs with more MAC counters fails are named.

C.1 Location of APs

Among the one hundred APs are part of the managed network, those distinguished by any characteristic are installed either in EETAC or Services building. Figures below show their exact location.

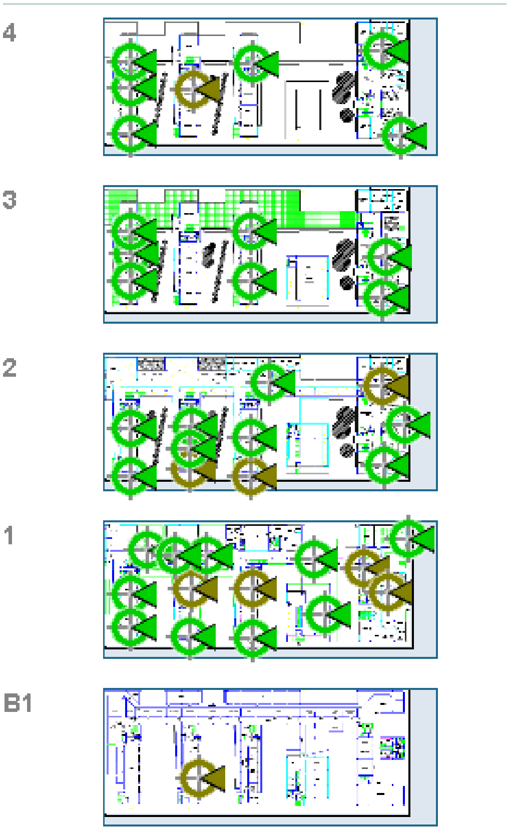


Fig. C.1. EETAC's elevation view

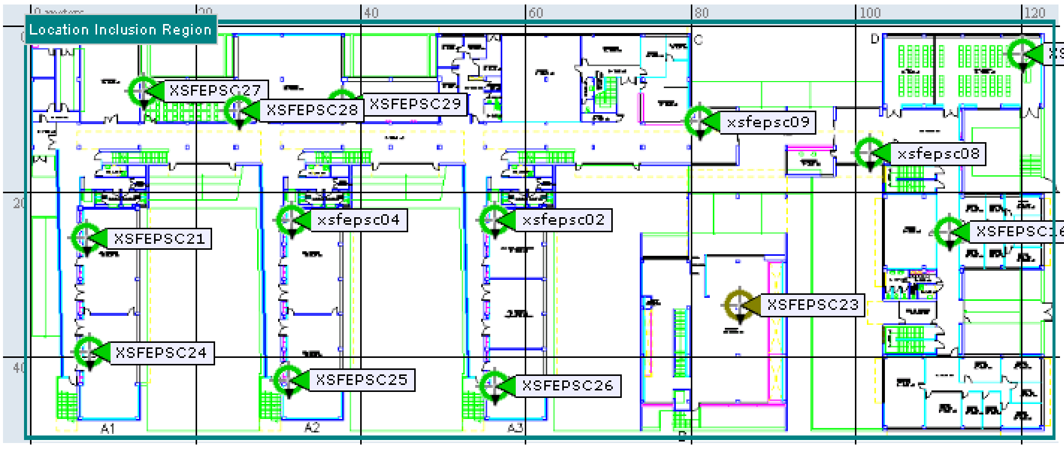


Fig. C.2. EETAC's ground floor

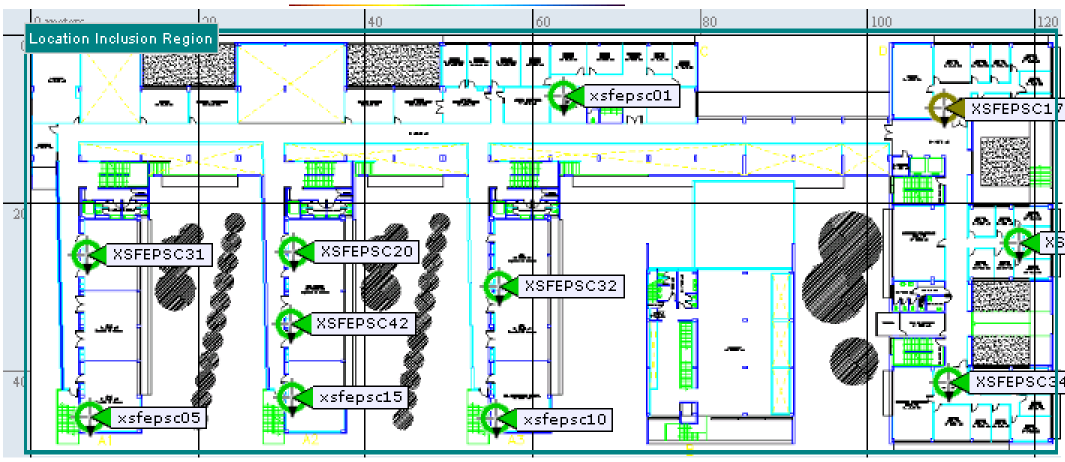


Fig. C.3. EETAC's first floor

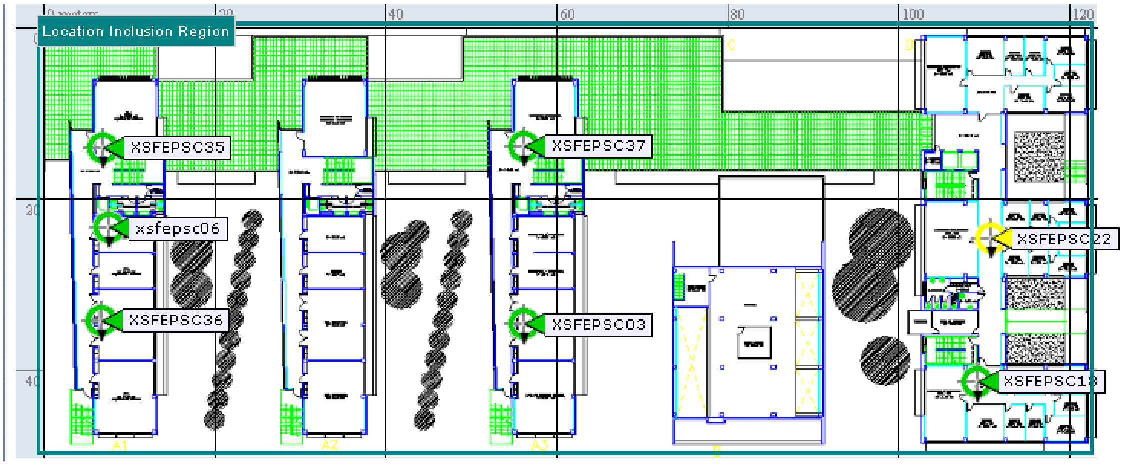


Fig. C.4. EETAC's second floor

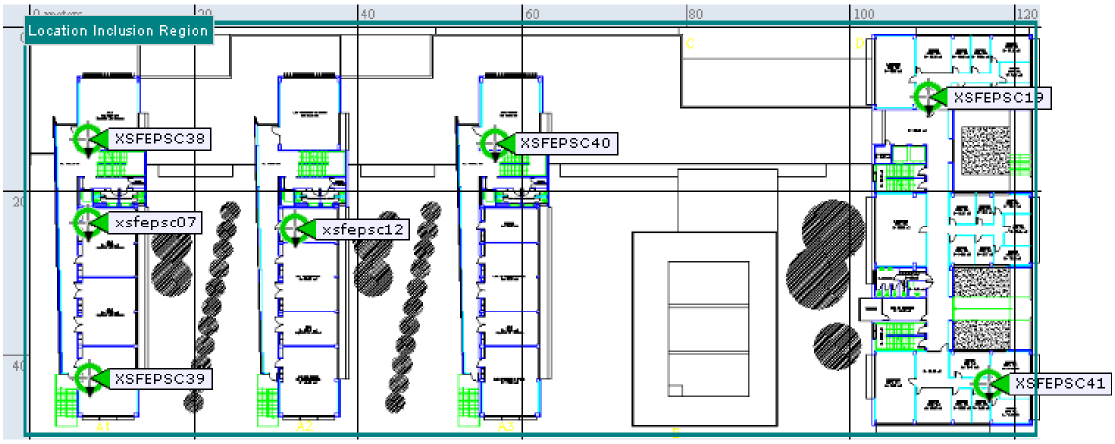


Fig. C.5. EETAC's third floor

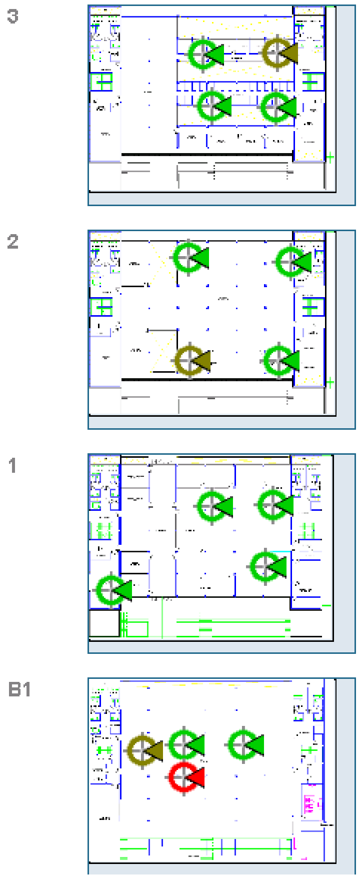


Fig. C.6. Services Building's elevation view

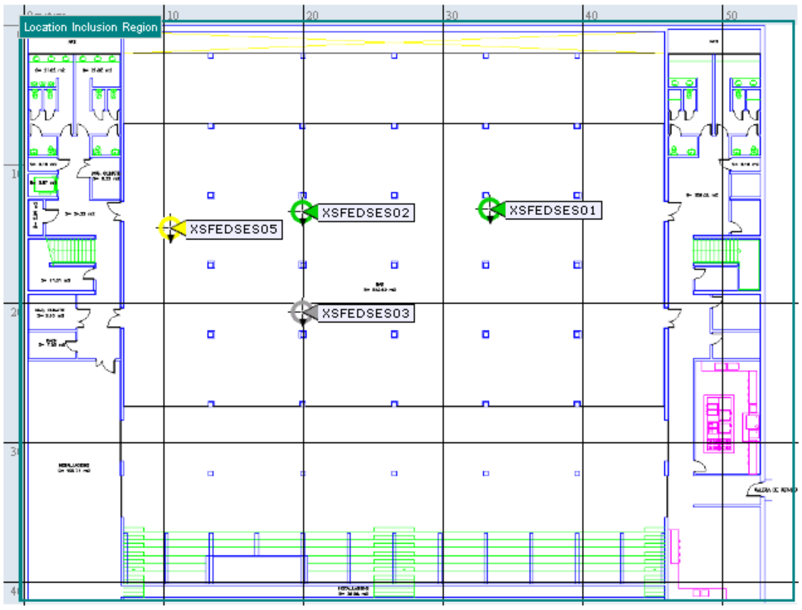


Fig. C.7. Services building’s underground floor (cantine)

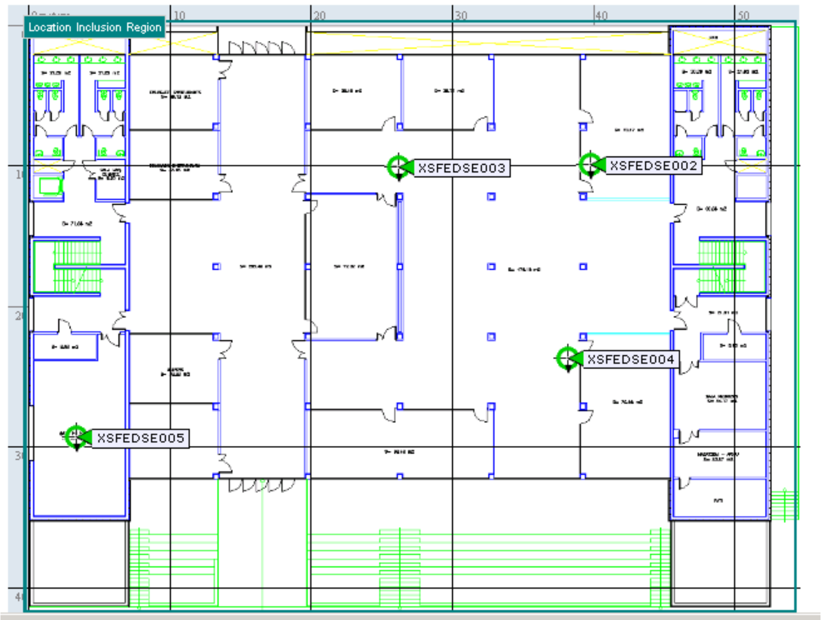


Fig. C.8. Services Building’s ground floor

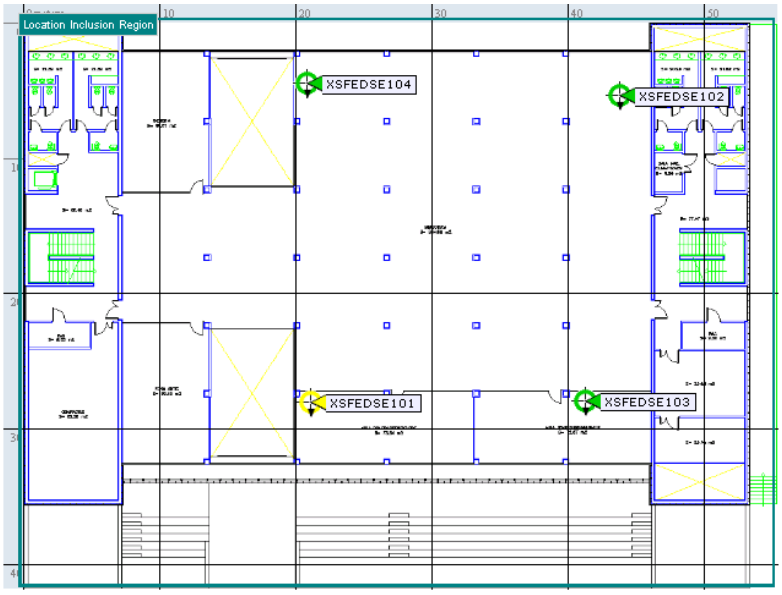


Fig. C.9. Services Building’s first floor (library)

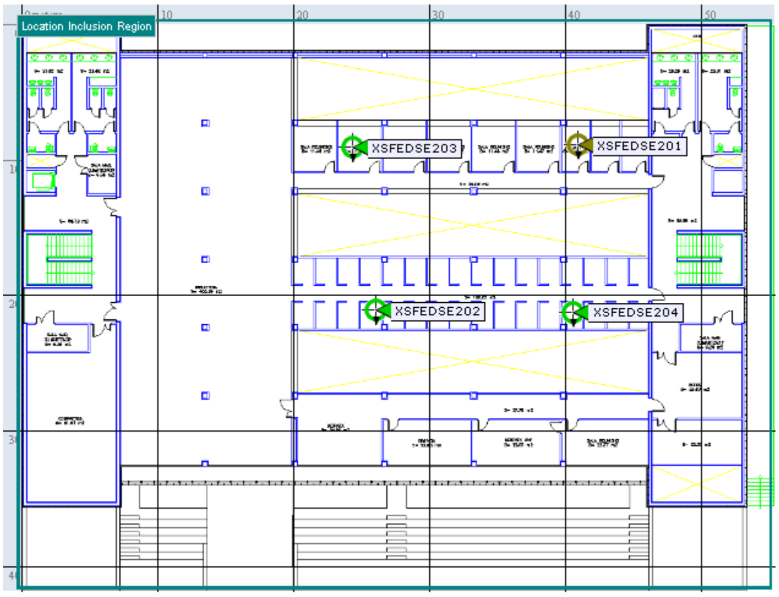


Fig. C.10. Services Building’s second floor (library)