WLAN - Project 2

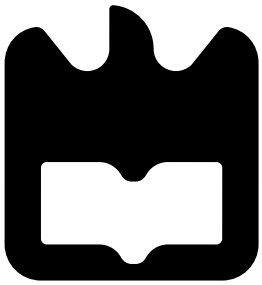
Comunicações Móveis

Universidade de Aveiro

Carlos Costa, Diogo Correia, João Simões

Group 2

2022/2023



WLAN - Project 2

Departamento de Eletrónica, Telecomunicações e Informática
Universidade de Aveiro

Carlos Costa (88755) - carlospalmacosta@ua.pt,
Diogo Correia (90327) - diogo.correia99@ua.pt,
João Simões (88930) - jtsimoes@ua.pt

January 6, 2023

Contents

Acronyms

AP Access Point.

BSSID Basic Service Set Identifier.

CLI Command Line Interface.

CTS Clear To Send.

dBm deciBel-milliWatts.

DHCP Dynamic Host Configuration Protocol.

GHz GigaHertz.

IEEE Institute of Electrical and Electronics Engineers.

ISP Internet Service Provider.

MAC Medium Access Control.

Mbps Megabits per second.

PHY Physical Layer.

RTS Request To Send.

SSID Service Set Identifier.

TCP Transmission Control Protocol.

UDP User Datagram Protocol.

VLAN Virtual Local Area Network.

VM Virtual Machine.

WLAN Wireless Local Area Network.

List of Figures

List of Tables

Chapter 1

Introduction

Institute of Electrical and Electronics Engineers (IEEE) 802.11 is part of the IEEE 802 set of technical standards and specifies the set of Medium Access Control (MAC) and Physical Layer (PHY) protocols for implementing Wireless Local Area Network (WLAN) communication. The standard's basic version was released in 1997, and further revisions have been made. The technology behind 802.11 is commonly branded to consumers as Wi-Fi. IEEE 802.11 uses various frequencies including, but not limited to, 2.4 GHz and 5 GHz. These are the frequencies we use in the course of this project.

With this project, we intend to test and verify the functional and performance aspects of IEEE 802.11 technology. More specifically, we will explore more about **802.11n** (Wi-Fi 4 generation), the version of the standard used by the equipment we were provided with. For this, we analysed the impact of coverage and performance configurations (addressed in Chapter 3), the occurrence of roaming between APs (addressed in Chapter 4), the ad-hoc mode (addressed in Chapter 5) and the importance of Request To Send (RTS) and Clear To Send (CTS) mechanism (addressed in Chapter 6).

Throughout this report, we will address each one of these situations, show our approach, the setup made and finally we will present our results and conclusions.

All the archives regarding the work developed can be found in the following GitHub repository: `https://github.com/digas99/cm-project-wlan2`

Chapter 2

Cisco Access Point

For this project, we were provided with two Cisco Access Points.

Model: AIR-CAP3602I-E-K9

(a) Top view (b) Bottom view

Figure 2.1: Access Point

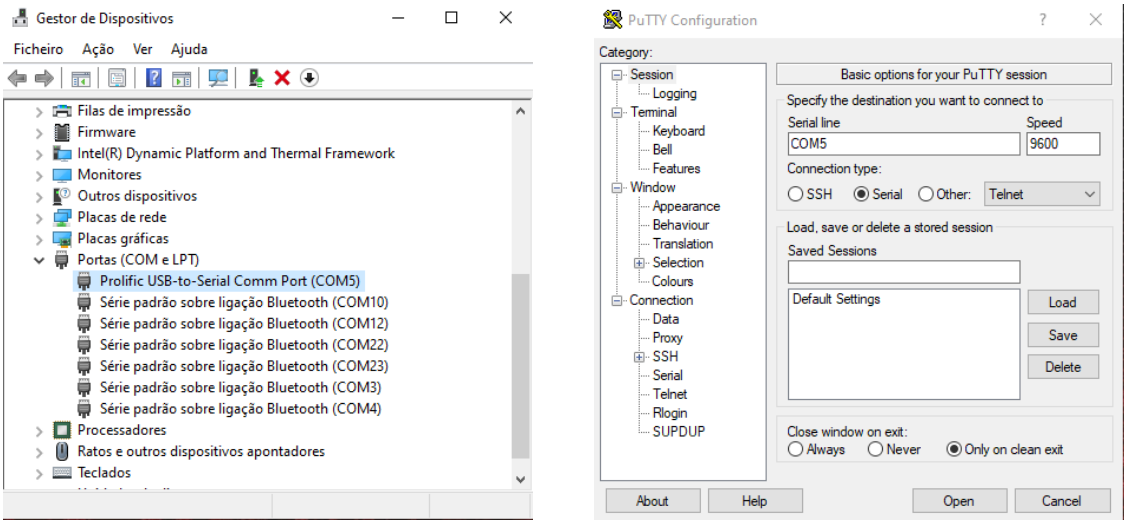
The images above were taken from the tonitrus.com website [1].

2.1 Getting started

To access the AP's console and make modifications to the configuration, we need a Console Cable (ethernet male to usb male). Upon connecting the AP to the computer, we need to identify the physical port it is connected to, and pass it to PuTTY [2] through a Serial connection type (Figure 2.2(b)). To identify to which port the AP is connected, we can use the Device Manager in Windows (Figure 2.2(a)).

In Linux, we can check the lastly connected port using `dmesg` [3], with the command:

```
> dmesg | grep USB
```



(a) Check Physical Port in Device Manager

(b) Open Console through PuTTY

Figure 2.2: Steps to Open AP Console

In PuTTY, upon clicking *Open*, a terminal should open. To enter configuration mode, we must use the command `enable` [4] and type the password, which is "Cisco" by default.

Figure 2.3: Console of the Access Point

Chapter 3

802.11 coverage and performance

A network connection may behave differently depending on a number of factors. A way of measuring this connection is by looking at the strength of its signal, and the rate of data transferred through such connection, the bandwidth. These two properties will be affected by several physical factors, such as the distance between the communicating entities and the obstacles between them. Having this in mind, 5 scenarios will be considered:

- **Scenario 1** - Device as close as possible to the AP (< 1m)
- **Scenario 2** - Device within sight of the AP (± 5m)
- **Scenario 3** - Device on another room, with a wall as an obstacle (± 5m)
- **Scenario 4** - Device on another end of the house, several walls as obstacles (± 10m)
- **Scenario 5** - Device on a different floor (± 15m)

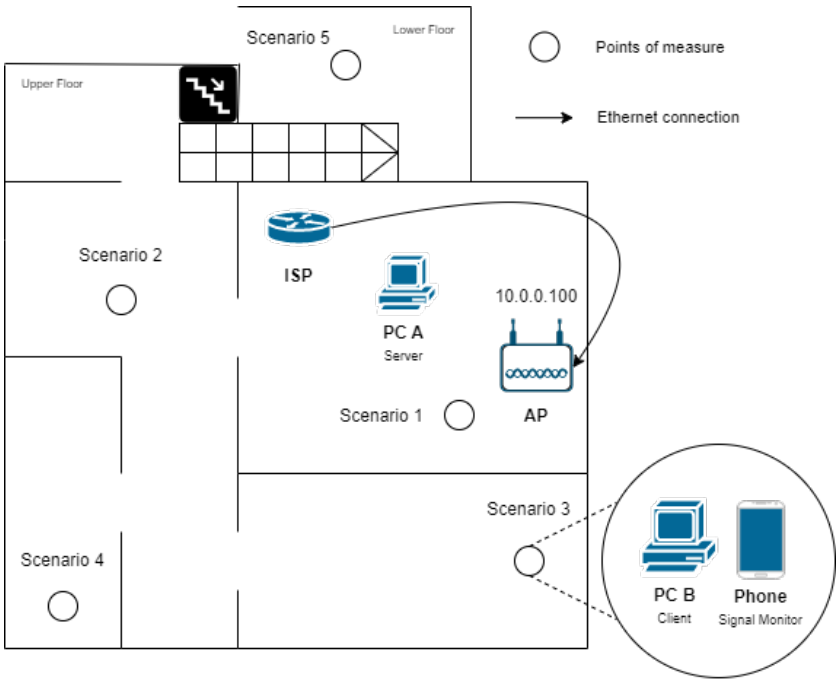


Figure 3.1: Blueprint illustrating all the different scenarios

3.1 Approach

To evaluate the performance of the IEEE 802.11, we will take into consideration the properties under the scenarios mentioned above, for the two most used frequency bands, 2.4GHz and 5GHz.

To measure the **signal strength**, we will use the mobile application *WiFi Analyzer* [5]. This app detects every SSID within a certain range and estimates a value for the signal strength, in deciBel-milliWatts (dBm). This tests were performed using version 3.5 on Android 10.

To measure the **bandwidth**, *iPerf3* [6] was used, which is a cross-platform tool that can produce standardized performance measurements for any network. It allows the setup of a server, and have clients send data to it. This open-source software then provided us with how much data was sent each second, for a set number of seconds (in this case, 20 seconds). This tests were performed using version 3.1.3 on Windows 10 (64-bit) via the Command Line Interface (CLI).

It was decided to put a fixed time limit of 20 seconds and check what was the maximum amount of data transferred from the client to the server in that time interval, but we could also have put a fixed data stream size and see how long it took until all the information was transferred. The data was transferred using Transmission Control Protocol (TCP).

Below, we show an example of the output that *iPerf* provided:

```
> iperf3 -c 10.0.0.2 -t 20 --logfile scenario1-test1-5g.txt

Connecting to host 10.0.1.4, port 5201
[  5] local 10.0.1.5 port 63897 connected to 10.0.1.4 port 5201
[ ID] Interval           Transfer     Bandwidth
[  5]   0.00-1.01   sec  4.38 MBytes  36.3 Mbits/sec
[  5]   1.01-2.01   sec  4.00 MBytes  33.8 Mbits/sec
[  5]   2.01-3.01   sec  3.88 MBytes  32.5 Mbits/sec
[  5]   3.01-4.00   sec  4.38 MBytes  36.8 Mbits/sec
[  5]   4.00-5.00   sec  4.25 MBytes  35.7 Mbits/sec
[  5]   5.00-6.01   sec  4.38 MBytes  36.2 Mbits/sec
[  5]   6.01-7.01   sec  4.38 MBytes  36.7 Mbits/sec
[  5]   7.01-8.01   sec  4.38 MBytes  36.7 Mbits/sec
[  5]   8.01-9.01   sec  4.12 MBytes  34.6 Mbits/sec
[  5]   9.01-10.01  sec  4.25 MBytes  36.0 Mbits/sec
[  5]  10.01-11.00  sec  4.38 MBytes  36.8 Mbits/sec
[  5]  11.00-12.00  sec  4.12 MBytes  34.6 Mbits/sec
[  5]  12.00-13.00  sec  4.12 MBytes  34.7 Mbits/sec
[  5]  13.00-14.00  sec  4.00 MBytes  33.4 Mbits/sec
[  5]  14.00-15.00  sec  4.12 MBytes  34.8 Mbits/sec
[  5]  15.00-16.01  sec  4.25 MBytes  35.3 Mbits/sec
[  5]  16.01-17.01  sec  4.25 MBytes  35.5 Mbits/sec
[  5]  17.01-18.00  sec  4.12 MBytes  35.0 Mbits/sec
[  5]  18.00-19.01  sec  4.25 MBytes  35.4 Mbits/sec
[  5]  19.01-20.01  sec  4.25 MBytes  35.8 Mbits/sec
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bandwidth
[  5]   0.00-20.01  sec  84.2 MBytes  35.3 Mbits/sec                  sender
[  5]   0.00-20.01  sec  84.1 MBytes  35.2 Mbits/sec                  receiver

iperf Done.
```

The most relevant value for us is the sender's bandwidth, present in the final summary of the command execution.

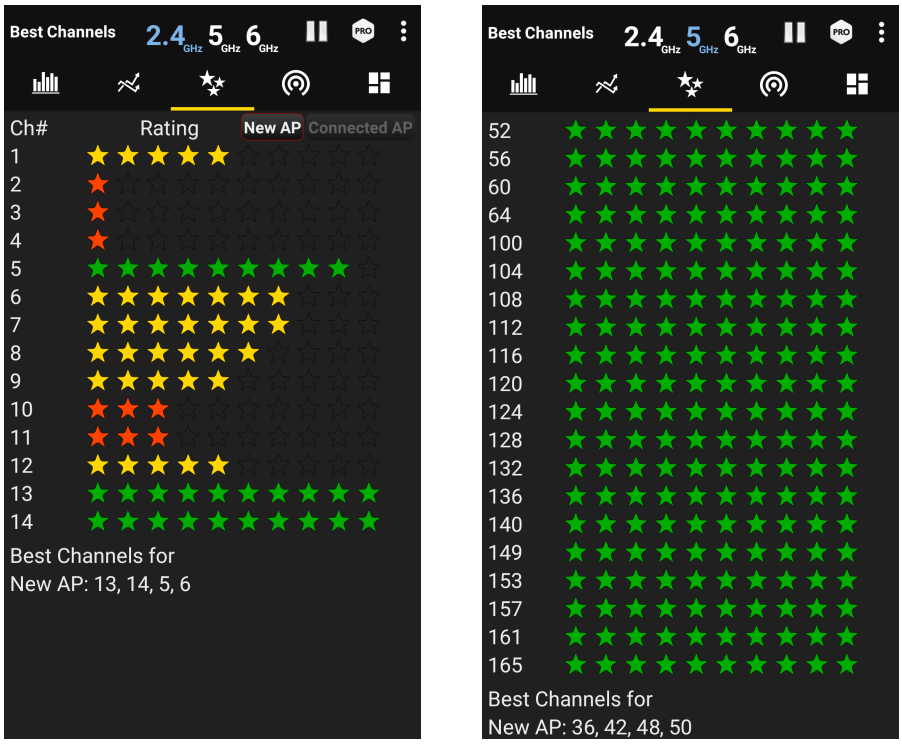
3.2 Setup

To perform this experiment, we will create a Virtual Local Area Network (VLAN) to conduct traffic in the 2.4GHz and 5GHz frequency bands. This VLAN will be created in the Access Point (AP), which is

directly connected through an Ethernet Cable to the ISP's Router to open the VLAN to the Internet.

Since we were in an apartment, there were a lot of networks and traffic going on, especially in the 2.4GHz band. So it was necessary to adjust the channel on which our AP would communicate on, in order to get the least interference from external factors in our performance/coverage tests. To help us in this task, we also used the *WiFi Analyzer* app, which contains a tab to inform the user what are the best choices for channels when placing a new AP.

In the case of 2.4GHz, there were about 25 Wi-Fi networks that were mainly using channel 2, 3, 4, 10 and 11, as can be seen in the figure Figure 3.2-a). Thus, we decided to follow the application's suggestion and assign channel 13 to our AP with respect to this frequency. In the case of 5GHz, since there were only 3 Wi-Fi networks using channel 40, the range of best choices for the AP's channel was large and we decided to use the first one the application indicates (Figure 3.2-b)) out of the many possibilities we had: channel 36.



(a) 2.4GHz (b) 5GHz

Figure 3.2: Analysis of the best channel for the AP configuration

Afterwards, we prepared 3 devices (2 PCs and 1 Mobile Phone) and connected them to our network, through two different SSIDs (one for each Wi-Fi interface), in order to make the performance tests taking into consideration the two 802.11 frequencies. PCs will be used to measure the bandwidth, one acting as the Server, the other as the Client. Finally, the Mobile Phone will be used just to measure the signal strength.

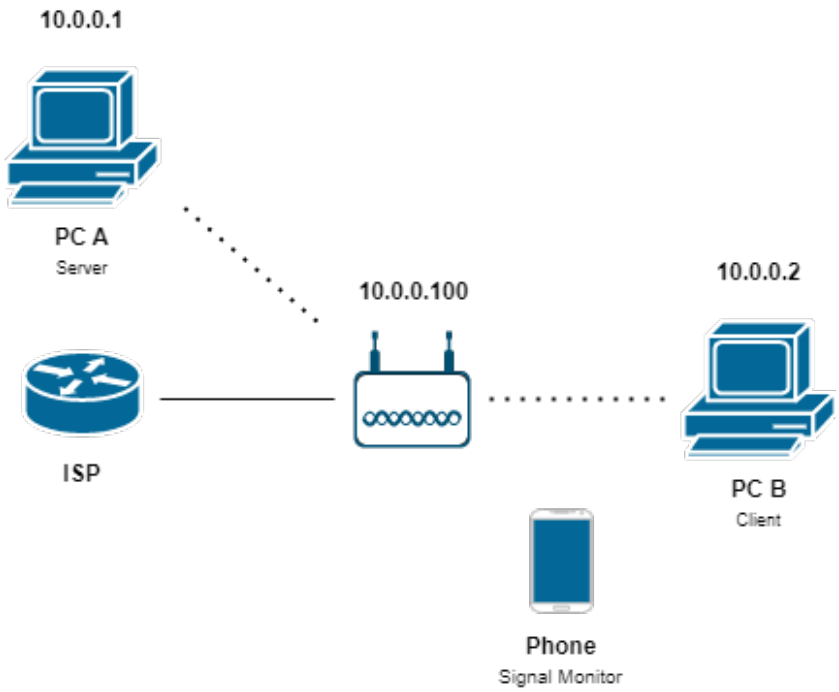


Figure 3.3: Setup for the measuring of the 802.11 performance

3.3 AP Configuration

```
1  enable
2
3  configure terminal
4
5    ! physical interface for wifi - 1 is for 5GHz
6    interface Dot11Radio 1
7      channel 36
8
9      ssid CMProject-5G
10     mbssid
11
12     rts threshold 200
13
14     no shutdown
15   exit
16
17   ! physical interface for wifi - 0 is for 2.4GHz
18   interface Dot11Radio 0
19     channel 13
20
21     ssid CMProject
22     mbssid
23
24     rts threshold 200
25
26     no shutdown
27   exit
```

```
28
29      dot11 ssid CMProject
30        authentication open
31        vlan 1
32        mbssid guest-mode
33      exit
34
35      dot11 ssid CMProject-5G
36        authentication open
37        vlan 1
38        mbssid guest-mode
39      exit
40
41      ! virtual wifi interface in vlan 1 for 2.4GHz
42      interface Dot11Radio 0.1
43        encapsulation dot1q 1
44        bridge-group 1
45      exit
46
47      ! virtual wifi interface in vlan 1 for 5GHz
48      interface Dot11Radio 1.1
49        encapsulation dot1q 1
50        bridge-group 1
51      exit
52
53      ! ethernet interface to allow for internet connection for vlan 1
54      interface GigabitEthernet 0.1
55        encapsulation dot1q 1
56        bridge-group 1
57      exit
58
59      ! bvi = bridge virtual interface
60      ! this interface is used to bridge wireless devices
61      interface bvi 1
62        ip address 10.0.0.100 255.255.255.0
63        no shutdown
64      exit
65
66      ! setup dhcp
67      service dhcp
68
69      ip dhcp excluded-address 10.0.0.100 10.0.0.254
70      ip dhcp pool cmproject
71        network 10.0.0.0 255.255.255.0
72        default-router 10.0.0.100
73      exit
74
75    end
76
77    write
78
79    disable
```

3.4 Results

3.4.1 Signal Strength

As mentioned on Section 3.1, to measure the signal strength, we used an Android smartphone with the *WiFi Analyzer* [5] app installed. We went through the 5 different scenarios/locations, waited for the signal to stagnate and then noted the value of the signal strength (in dBm), for both 2.4 and 5 GHz band.

The readings for each scenario were as follows:

Figure 3.4: Signal strength test results for Scenario 1

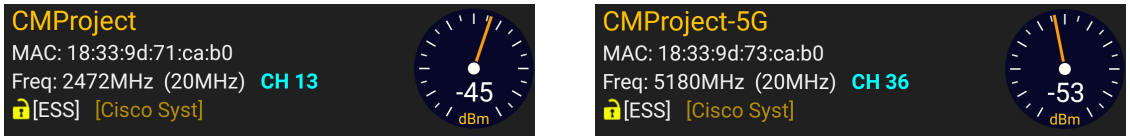


Figure 3.5: Signal strength test results for Scenario 2

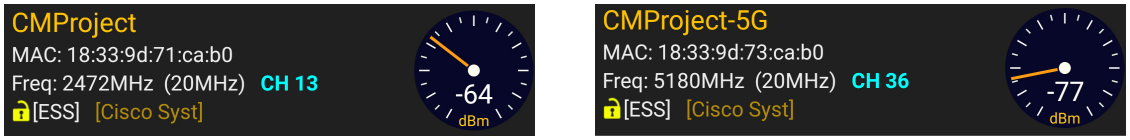


Figure 3.6: Signal strength test results for Scenario 3

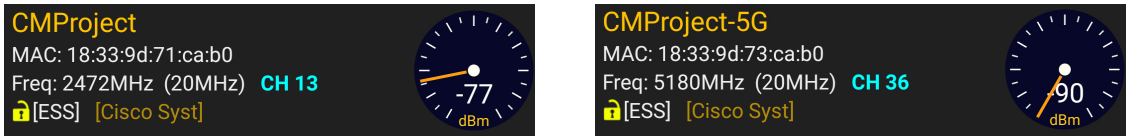


Figure 3.7: Signal strength test results for Scenario 4

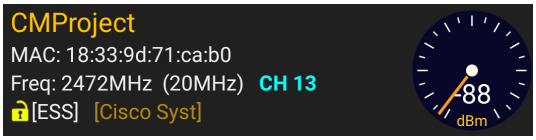


Figure 3.8: Signal strength test results for Scenario 5 (5GHz network not reachable in this case)

The bar chart below (Figure 3.9) summarizes the signal strength values read in the five different scenarios for both the 2.4 and 5 GHz band. For a better visualization, the graph is inverted, since the values are negative. Thus, a shorter bar means a better signal, a longer bar means a worse signal.

Note that a bar is missing in Scenario 5 for the 5GHz series, since the network was not reachable on the mobile phone at that distance.

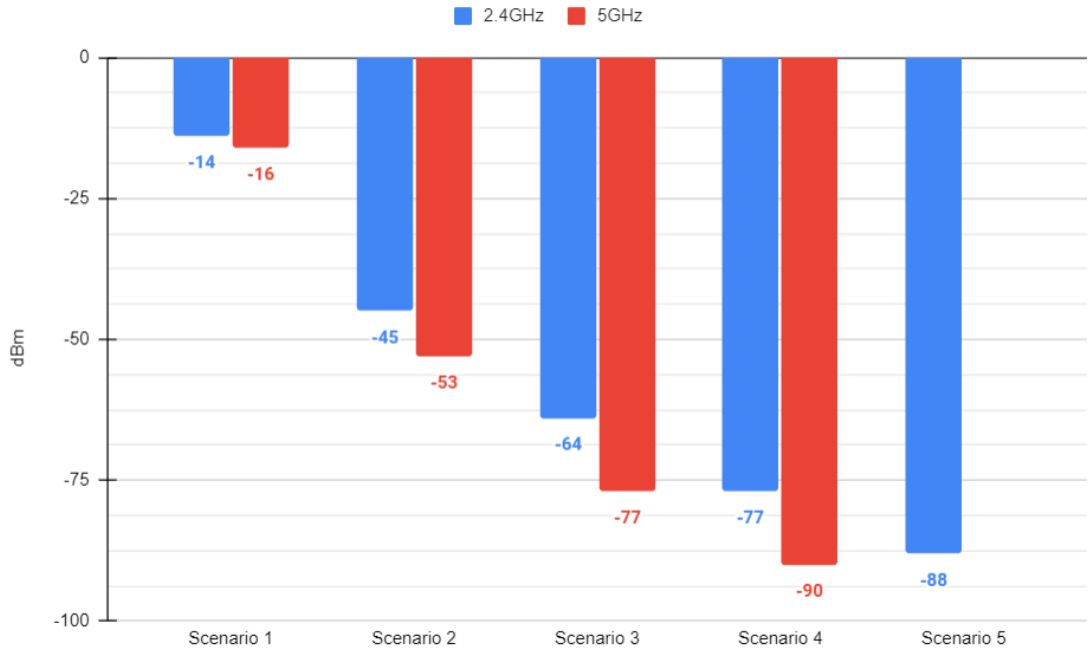


Figure 3.9: Side-by-side comparison of the signal strength test results on both bands for each scenario

The results match the predictions we had made prior to starting the tests. As we increase the distance and obstacles in relation to the AP, the signal gets worse and worse, until the network is no longer reachable. As expected, in the scenario closest to the AP (scenario 1), we get the best possible signal quality ($<$ 20 dBm) and in the scenario farthest from the AP (scenario 5) we get the worst possible signal quality ($>$ 80 dBm) and even signal loss entirely, in the case of the 5GHz network.

One notable difference between the two bands is that the 5GHz band is more affected by distance and obstacles than the 2.4GHz band. At first (when close to the AP), the difference in the signal strength between the two bands is almost insignificant, but as we move further away from the AP, the difference becomes more noticeable. This is due to the fact that 5GHz frequency has a smaller coverage area and is worse at penetrating solid objects than the 2.4GHz one.

3.4.2 Bandwidth

As mentioned on Section 3.1, to measure the bandwidth, we used two Windows PC's with the *iPerf* [6] software installed: one to function as a server (command `iperf -s`), the other as a client (command mentioned in Section 3.1). We placed the server PC near the AP, then we performed 5 times on each scenario/location the transfer of as much data as possible, during 20 seconds, from the client to the server and then recorded the bandwidth value we were presented with on the terminal (in Mbps), for both 2.4 and 5 GHz band.

All the readings for each scenario were as follows:

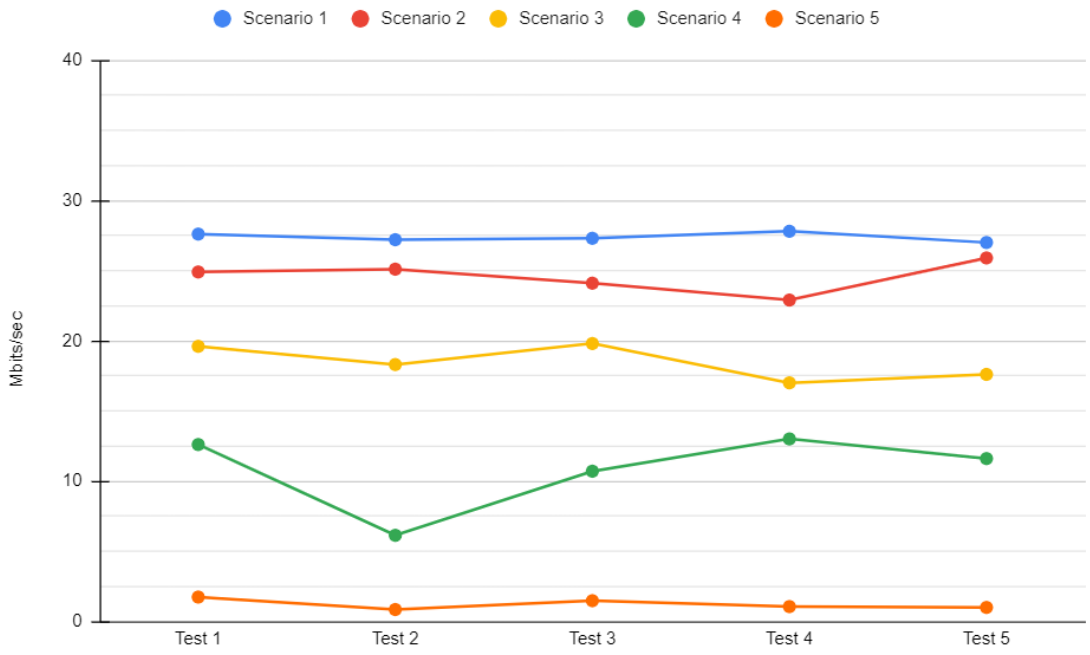


Figure 3.10: Bandwidth test results on 2.4GHz for each scenario

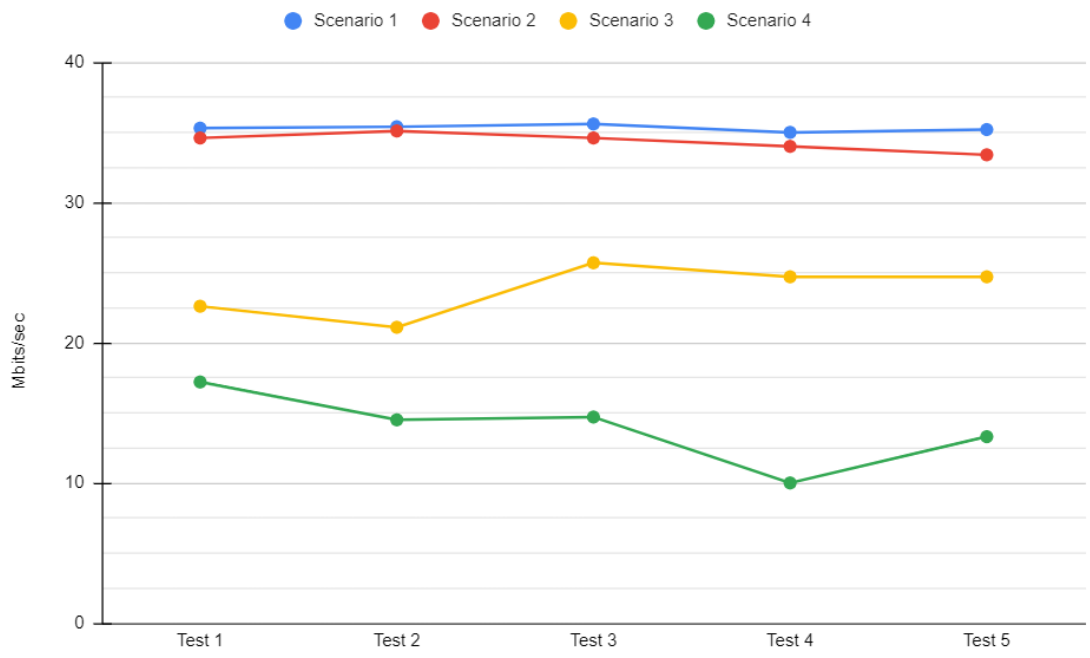


Figure 3.11: Bandwidth test results on 5GHz for each scenario (5GHz network not reachable in scenario 5 case)

The bar chart below (Figure 3.12) summarizes all the bandwidth values read in the five different scenarios for both the 2.4 and 5 GHz band. For a better visualization and to decrease the dispersion between the

read values, we only present the average bandwidth across the five tests. A shorter bar means a lower bandwidth, a longer bar means a higher bandwidth.

Again, note that a bar is missing in Scenario 5 for the 5GHz average series, since the network was not reachable on the Client PC at that distance.

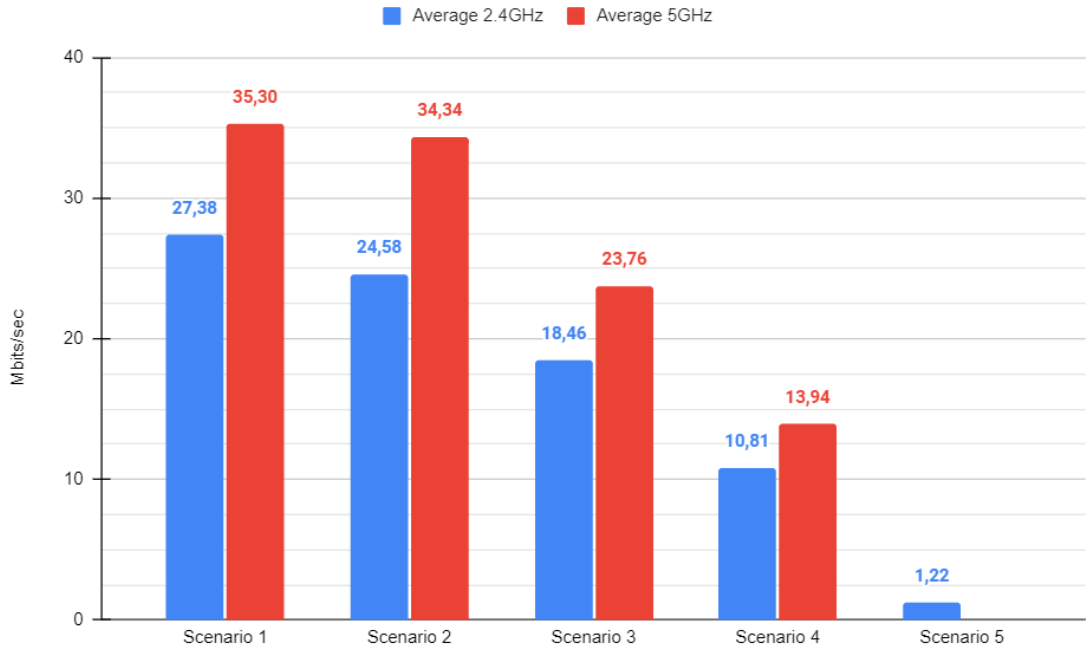


Figure 3.12: Side-by-side comparison of the average bandwidth test results on both bands for each scenario

All the detailed information about each of the tests performed can be found in the project's repository, at the following link: `https://github.com/digas99/cm-project-wlan2/tree/main/results/bandwidth`.

Once again, the results match the predictions we had made prior to starting the tests. As we increase the distance and obstacles in relation to the AP, the bandwidth gets worse and worse, until the network is no longer reachable. As expected, in the scenario closest to the AP (scenario 1), we get the higher bandwidth and in the scenario farthest from the AP (scenario 5) we get the lowest bandwidth and, consistent with the signal strength results (Subsection 3.4.1), even signal loss entirely, in the case of the 5GHz network.

However now, the difference in the average bandwidth between the two bands is more substantial in the case where we are close to the AP than when we move further away from it. This is due to the fact that 5GHz frequency has a higher data rate and is less prone to interference than the 2.4GHz, which makes reach higher transfer speeds the closer we are to the server. Nevertheless, recalling what we discussed in the previous test results, the 2.4GHz has larger coverage area, losing less signal strength and therefore less bandwidth than 5GHz, as we move away from the AP.

Given all these pros and cons of the two bands, for a better Wi-Fi performance, we should analyse where and for what we will use our connection most.

For example, if we want a longer range for our devices (e.g.: devices that moves around a lot throughout the day, like mobile phones, baby monitoring equipment, PCs for simple internet surfing, etc.), using 2.4GHz is the best option.

On the other hand, if we need a higher speed for high-bandwidth activities online (such videoconferencing, gaming or high-definition streaming) and if we can sacrifice range, then 5GHz is most likely the better option. Better yet for this case, would be to connect directly to the router/AP with an Ethernet

cable (IEEE 802.3), which is already out of the scope of this project, but, being a wired connection, it will certainly always be more stable and faster than the wireless connection.

Another important case to look at is if we are in a flat or condo with a lot of interference from other Wi-Fi networks around it, 5GHz will help us avoid wireless congestion, as it's less susceptible to overlap from other devices.

Chapter 4

Roaming between APs

Wireless connections can only go as far as its signal's preponderance through the air. Because the signal transmitted from the antenna gets weaker with distance (the electromagnetic wave spreads, and so does its power), a building offering a connection through a WLAN might not be able to offer full coverage within its precinct. To spread the connection further, with a decent strength, Roaming is used. This is a technique implemented by devices (users of the network) to hop from router to router, so that the user might always achieve the best connectivity between all routers announcing a same network.

4.1 Approach

We will analyse the handover process of the Roaming mechanism in action by measuring the signal strength and distance between the two APs and a PC.

For the distance between the APs we will make an approximation. To measure the signal strength we will use the mobile application *WiFi Analyzer* [5].

Because at 2.4GHz the radio waves travel further with less strength, as opposed to waves in the 5GHz band, which have more strength at a shorter range, we will perform the same measurements for those two frequency bands. For this, we will create two SSIDs in each Radio Interface of the AP.

Near the AP 1, we will begin by connecting the PC (a laptop) to one of the SSIDs, register the signal strength and walk slowly towards AP 2. We keep walking until the laptop connects to AP 2, and when that happens we register again the signal strength and our distance to AP 1.

Because both APs announce the same network (same SSID), we will need a way to understand when the handover occurs. For this, two methods can, and will be used, which are the following:

- LED on top of the AP (Figure 4.1)
- Network Shell command: `netsh wlan show interfaces` [7]

Figure 4.1: Led on top of the AP

As per the Cisco Instruction Manual [8] for the AP in use, a green light means a "Normal operating condition, but no wireless client associated" and a blue light means a "Normal operating condition, at least one wireless client association". With this in mind, we can see, as the led s from green to blue in the AP 2, when our laptop performs the handover. At the same time, we can continuously run the Network Shell command on the laptop and watch out for a in the BSSID, which is the MAC physical address of the AP's radio interface.

We have, beforehand, written down all the BSSIDs we needed, to then be able to compare and identify the measurements.

Access Point	Interface	SSID	BSSID
AP 1	Dot11Radio 0	CMProject	18:33:9d:c7:b0:b0
	Dot11Radio 1	CMProject-5G	18:33:9d:c3:b1:10
AP 2	Dot11Radio 0	CMProject	18:33:9d:71:ca:b0
	Dot11Radio 1	CMProject-5G	18:33:9d:73:ca:b0

Table 4.1: BSSIDs for each connection

4.2 Setup

For this experiment, we will have both wi-fi interfaces of the AP connected to a VLAN. The bridge interface will have an IP in the same network as the ISP's router. This allows the AP, when directly connected, via ethernet, to the router, to relay the ISP's network (which is connected to the Internet). This setup will be done for two APs.

To connect to the network, we will use a PC that will receive an IP through DHCP. The Roaming Aggressiveness of the Network Adapter was set to Medium (default). This setting sets the threshold at which the Wi-Fi adapter starts scanning for other APs [9].

To measure the signal strength, we will use a mobile phone equipped with the already mention *WiFi Analyzer* application.

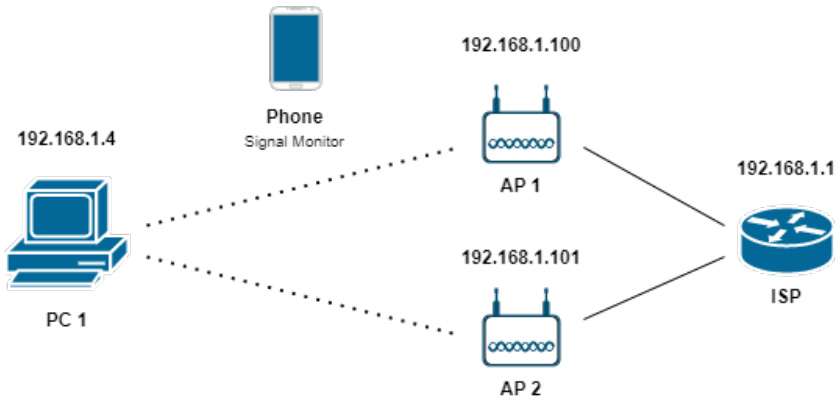


Figure 4.2: Setup to test Roaming

4.3 AP Configuration

```
enable

configure terminal

  ! physical interface for wifi - 1 is for 5GHz
  interface Dot11Radio 1
    channel 36

    ssid CMProject-5G
    mbssid

    rts threshold 200

    no shutdown
  exit

  ! physical interface for wifi - 0 is for 2.4GHz
  interface Dot11Radio 0
    channel 13

    ssid CMProject
    mbssid

    rts threshold 200

    no shutdown
  exit

  dot11 ssid CMProject
    authentication open
    vlan 1
    mbssid guest-mode
  exit

  dot11 ssid CMProject-5G
    authentication open
    vlan 1
```

```
38      mbssid guest-mode
39    exit
40
41    ! virtual wifi interface in vlan 1 for 2.4GHz
42    interface Dot11Radio 0.1
43      encapsulation dot1q 1
44      bridge-group 1
45    exit
46
47    ! virtual wifi interface in vlan 1 for 5GHz
48    interface Dot11Radio 1.1
49      encapsulation dot1q 1
50      bridge-group 1
51    exit
52
53    ! ethernet interface to allow for internet connection for vlan 1
54    interface GigabitEthernet 0.1
55      encapsulation dot1q 1
56      bridge-group 1
57    exit
58
59    ! bvi = bridge virtual interface
60    ! this interface is used to bridge wireless devices
61    interface bvi 1
62      ip address 192.168.1.100 255.255.255.0
63      no shutdown
64    exit
65
66  end
67
68  write
69
70  disable
```

4.4 Results

We began by testing on the 5GHz connection.

In the room, close to AP 1, we measured a signal strength of -51dBm, and a weaker signal of -79dBm from AP 2, which was more or less 9 meters away. We began walking slowly, with the laptop on our hands, through the house, continuously checking, both the LED on AP 2 and the output of the Network Shell command (Figure 4.3).

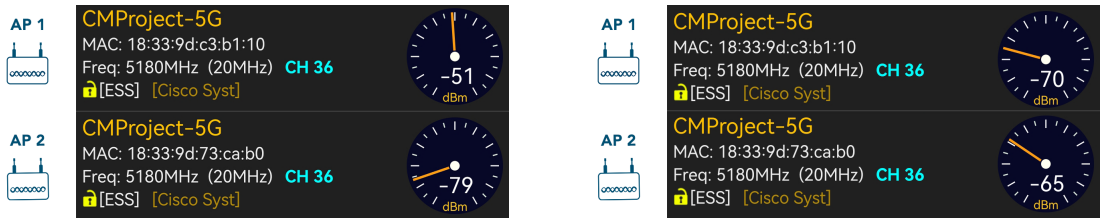
```
There is 1 interface on the system:

    Name                   : Wi-Fi
    Description            : Intel(R) Dual Band Wireless-AC 7265
    GUID                   : e7a4305c-2d14-438d-84d5-2fbef1c33a0c
    Physical address       : d0:57:7b:9f:c0:8b
    State                  : connected
    SSID                   : CMProject-5G
    BSSID                  : 18:33:9d:c3:b1:10
    Network type           : Infrastructure
    Radio type             : 802.11n
    Authentication         : Open
    Cipher                 : None
    Connection mode        : Profile
    Channel                : 36
    Receive rate (Mbps)    : 144.4
    Transmit rate (Mbps)   : 144.4
    Signal                 : 96%
    Profile                : CMProject-5G

    Hosted network status  : Not available
```

Figure 4.3: Network Shell command output example for one of the connections

At some point, at about 7 meters away from AP 1 (78% of the way), the handover was done. Here, the signal strength for AP 1 was a lower value of -70dBm, and a greater value of -65dBm for AP 2, when comparing to the first measurements, in the beginning (Figure 4.4).



(a) At the Beginning (b) At the Handover

Figure 4.4: Signal Strength of each AP during the experiment (5GHz)

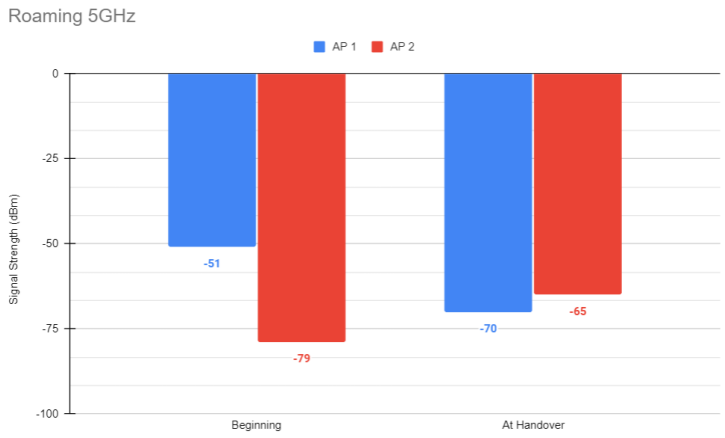
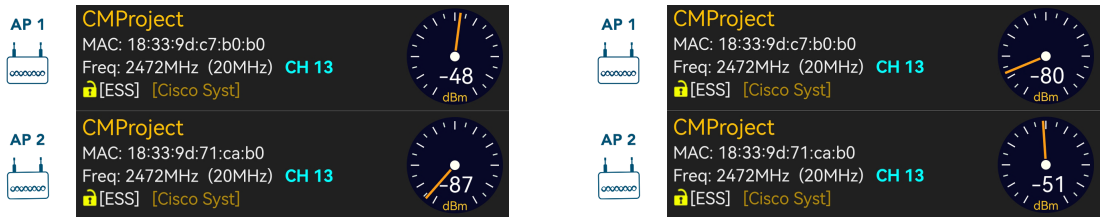


Figure 4.5: Bar chart with the Signal Strength of each AP during the experiment (5GHz)

Looking at the Figure 4.5, we can see that the handover happened when AP 1 had still a reasonable signal strength, just a bit lower than the strength of AP 2's.

For the 2.4GHz connection, we had more or less the same difference of signal strength in the beginning, as we had in the last experiment. A value of -48dBm for AP 1 and -87dBm for AP 2.

Keeping the same strategy as before, we started walking slowly, waiting for a change in the connection, but this time it didn't happen at 7 meters. Neither did it happen when we reached AP 2, at 9 meters. We actually had to go a couple of meters further, and only at 11 meters away from AP 1 (122% of the way and 2 meters ahead of the position of AP 2) the laptop changed its connection to AP 2. Here, the signal strength of AP 1 was a very low -80dBm, while the signal strength of AP 2 was a more reasonable one of -51dBm (Figure 4.6).



(a) At the Beginning (b) At the Handover

Figure 4.6: Signal Strength of each AP during the experiment (2.4GHz)

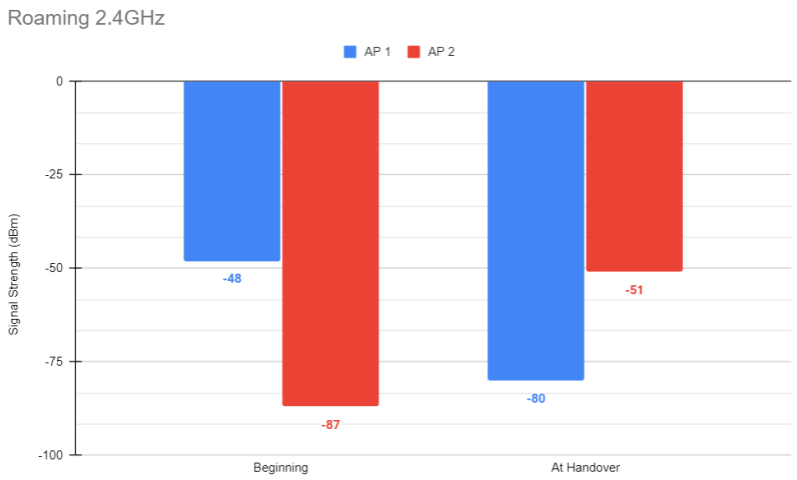


Figure 4.7: Bar chart with the Signal Strength of each AP during the experiment (2.4GHz)

Looking at the Figure 4.7, we can clearly see that, this time, with 2.4GHz, the connection held for longer than when we were using 5GHz. In the last experiment the handover happened at -70dBm, while this time it happened only at -80dBm.

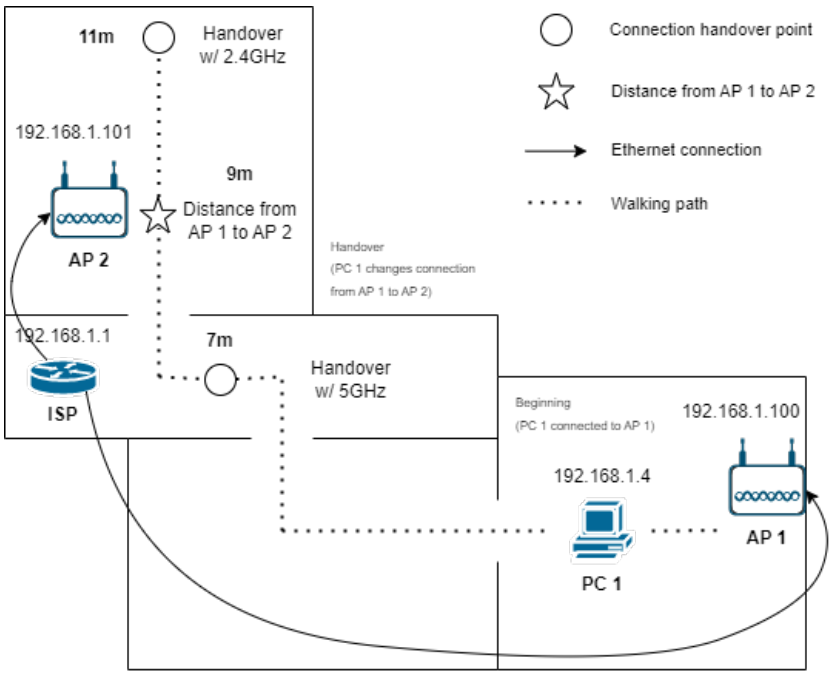


Figure 4.8: Blueprint showing where the handover process occurred

The blueprint at Figure 4.8 helps understand the positioning of the devices and the points where the handovers occurred. With 5GHz we were still between both APs at the time of the change, whereas at 2.4GHz we were beyond AP 2.

We also tried closing the wooden doors, represented by the gap on the walls in the blueprint above (2 doors until the 5GHz handover and 3 doors until the 2.4GHz), but it didn't make any significant difference.

There were other experiments, hinted by the professors at the time of the presentation of the project, that could have been made, but it didn't occur to us at the time.

One involved varying the aggressiveness of the Roaming in the laptop network driver and register how it would influence the process of changing connections from one AP to the other. A higher aggressiveness would trigger the connection handover while the signal strength of the current AP was still good. A lower aggressiveness would behave the opposite. If the APs are relatively close to one another, then having a lower aggressiveness would be problematic because by the time the current AP has a low signal strength, the other AP, even though slightly better than the first one, would have a low signal strength too. Having this in mind, a high aggressiveness is better for when the APs are closer to each other, and a low aggressiveness is better for when they are further apart.

Another experiment would be to calculate how long it would take for the laptop to connect to an AP, depending on the distance between them.

Chapter 5

Ad-hoc mode

An ad-hoc network is a type of wireless network that is set up on the fly, without the need for a central router or access point. It is a decentralized network in which the devices communicate directly with each other, rather than through a central server. Ad-hoc networks are often used in situations where it is not practical to set up a more traditional wired or wireless network, such as in a disaster recovery scenario, or when setting up a temporary network at a conference or event.[10]

In this section, we will see the differences between an ad-hoc network and an AP network.

5.1 Approach

To check these differences, we first connected two devices in an ad-hoc network and, with the help of *Wireshark*, see what kind of packages where exchanged between those devices. Next, we made the same tests of signal strength and bandwidth done to the AP network, to check how the network behaved differently.

5.2 Setup

5.2.1 Ad-hoc Network Setup

When setting up the ad-hoc network, we first needed to check if the device supported hosted networks by running the command on Windows:

```
> netsh wlan show drivers
```

The console should show then characteristics of the Wi-Fi driver of the device:

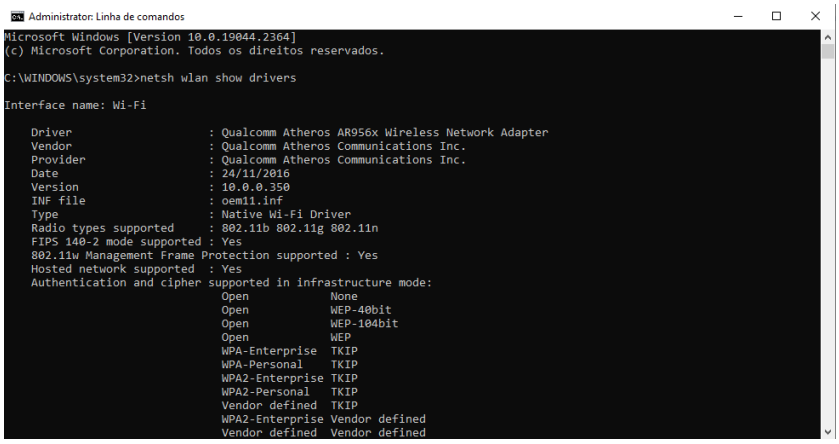


Figure 5.1: Details of the Wi-Fi driver

If the `Hosted network supported` gives a `Yes` output, then it means that the user can setup an ad-hoc network on the device. In our case, only one laptop supported this feature.

Next, to setup the ad-hoc network, we run the next command to create the network, where on the `ssid` field we will give the name of our network, and on the `key` field we will give its password:

```
> netsh wlan set hostednetwork mode=allow ssid=AdHocNetwork key=qwerty
```

Finishing up the ad-hoc network setup, we simply must start the network using the following command:

```
> netsh wlan start hostednetwork
```

5.2.2 Mode Monitor Setup

Setting up the devices to observe the packages shared between client and server, we followed this arrangement:

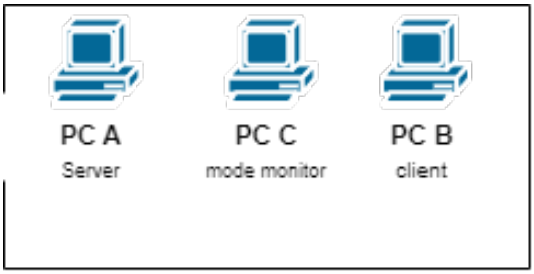


Figure 5.2: Monitor mode setup

5.2.3 Signal Strength

To test the signal strength of the ad-hoc network, we followed the same conditions as the signal strength test for the AP network using the same signal monitor.

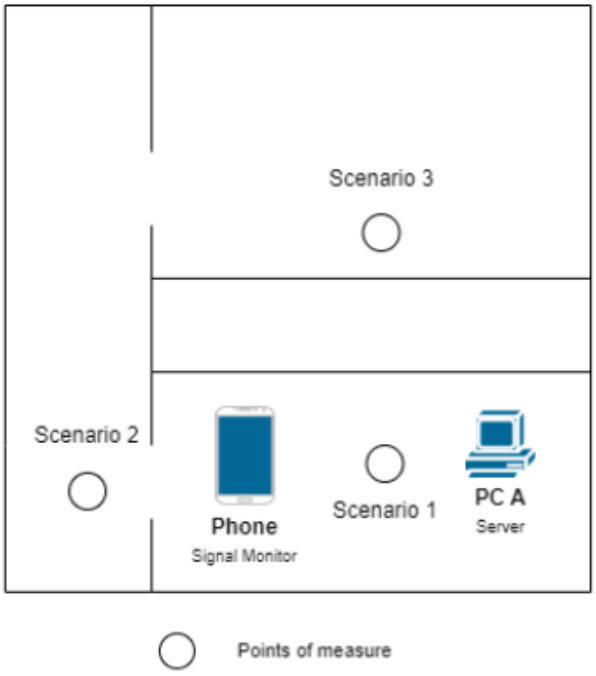


Figure 5.3: Signal strength test blueprint

5.2.4 Bandwidth

To test the bandwidth of the ad-hoc network, we followed the same conditions as the first scenario of the bandwidth tests of the AP network, while the client device sent *iPerf* data to the server.

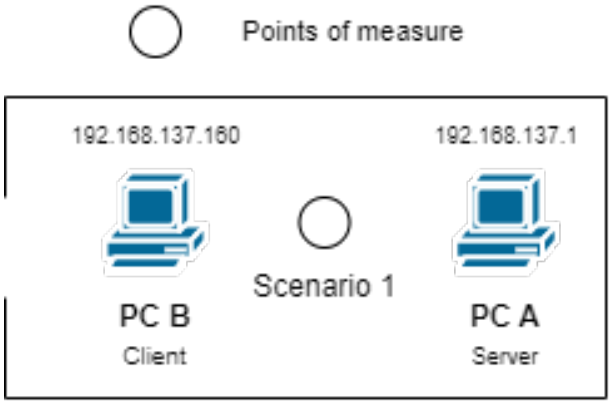


Figure 5.4: Bandwidth test blueprint

5.3 Results

5.3.1 Monitoring the packages

While the client device connected to the server, we checked on the device running *Wireshark* on monitor mode that there were packages outside the 802.11 protocol being exchanged. For example, on the figure below we can see that when the devices connected with each other, there where being sent LLS protocol packages encapsulated on 802.11 protocol packages.

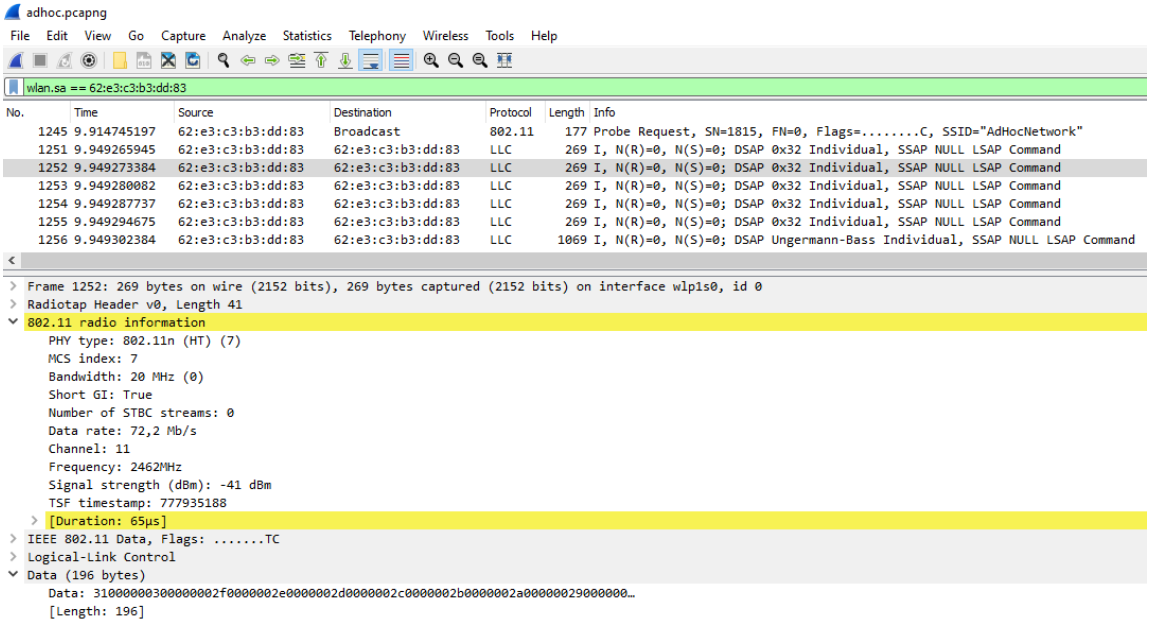


Figure 5.5: Packages captured with *Wireshark* when the client connected

This is possible because ad-hoc is a decentralized network, and it can permit any kind of packages to be exchanged outside of the 802.11 protocol. When the client device started doing *iPerf* tests to the server, however, it returned to send 802.11 protocol packages, as expected.

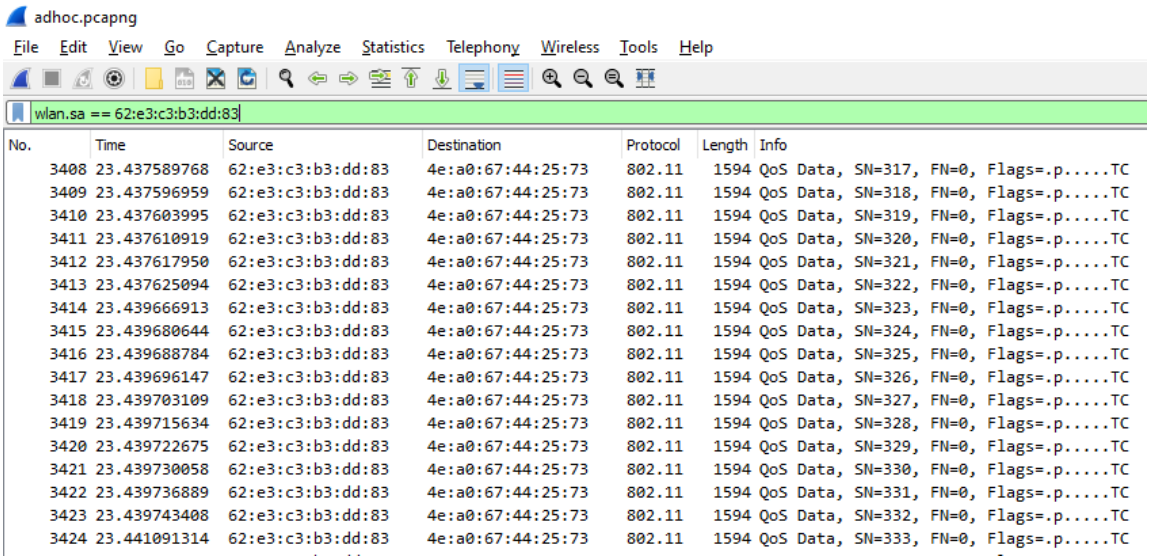


Figure 5.6: Packages captured with *Wireshark* while doing *iPerf* tests from client to server

5.3.2 Signal Strength

Similar to the AP network tests, we used the *WiFi Analyzer* app to measure the signal strength of the ad-hoc network. This time though, it was not possible to reach the signal beyond the scenario 3.

The results are as it follows:

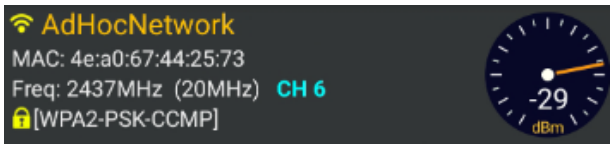


Figure 5.7: Signal strength on scenario 1

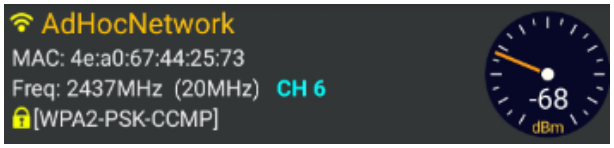


Figure 5.8: Signal strength on scenario 2

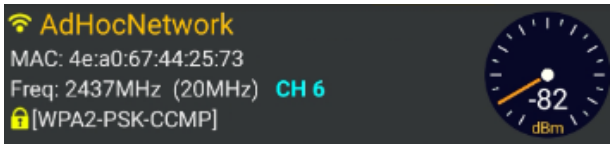


Figure 5.9: Signal strength on scenario 3

As we can observe, the signal strength of the ad-hoc network is weaker than the signal strength of the AP network (2.4GHz). The main reason is because it does not have a central access point. An ad-hoc network can have weaker signal strength than an infrastructure network that is set up with an access point, which can amplify the signal and provide a stronger connection.

Another reason has to do with the device limitations. The wireless capabilities of the device on an ad-hoc network can also affect the signal strength, as it depends of the device's network adapter.

5.3.3 Bandwidth

For the bandwidth test, we followed the same conditions of the scenario 1 of the bandwidth test for the AP network so we can compare both of the results.

The results are as it follows:

Scenario 1 **Device as close as possible to the AP (<1m)**	
AP **average bandwidth**	**Ad-hoc** **average bandwidth**
27.38 Mbits/sec	31.64 Mbits/sec

Table 5.1: Comparison between the bandwidth of AP network and ad-hoc network in scenario 1

As we can see, the bandwidth of the ad-hoc network is bigger than the bandwidth of the AP network. Because there is no central access point in an ad-hoc network, all devices in the network can communicate directly with each other, which can lead to a larger overall bandwidth compared to a network that relies on a central access point. In an AP network, all devices must communicate with the AP, which can act as a bottleneck and limit the overall bandwidth available to devices on the network.

Additionally, ad-hoc networks can potentially have a larger bandwidth because they can use the full capacity of each device's wireless network adapter, whereas an AP network can only use a portion of the available bandwidth, as some of it is used for communication with the AP and other network infrastructure.

Chapter 6

Control Frames (not accomplished)

We were going to test and analyse two control frames used to avoid collisions and perform a congestion control in the network traffic, Request To Send (RTS) and Clear To Send (CTS), but we did not manage to get it done.

The main approach would be to vary the RTS threshold and write down the behaviour of the mechanism under different circumstances. This threshold specifies the packet size from which the handshake should be done. Below that value an RTS/CTS handshake is not performed.

The different circumstances would consist of sending different packets, some TCP, some UDP, through a network with more or less traffic. To vary the amount of traffic in the network, we were going to use a WiFi Jammer [11], meant to flood the network, which would increase the probability of packet collisions. We failed to get the WiFi Jammer to work [12], so we lost the possibility to vary significantly the amount of traffic in the network.

The only tests we managed to make was seeing these control packets in action through pings and data transfers (with *iPerf*) from a client to a server, connected through an AP, as described in Figure 6.1.

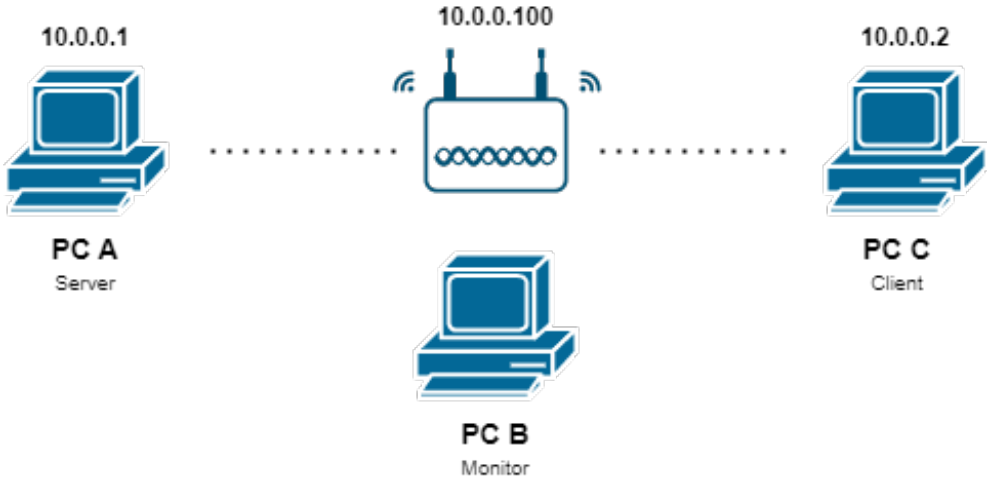


Figure 6.1: Setup for the analysis of the RTS/CTS behaviour

In both tests, we varied the size of the data sent in the packets and the threshold of the RTS in the AP config, using the rule `rts threshold [value from 0 to 2347]`. We expected to see an absence of RTS and CTS packets whenever the threshold was lower than the size of the packets but, for some reason, this didn't happen. The handshake was always done, no matter the size of the packets being sent.

Because of the noise captured by PC B, while monitoring packets on *Wireshark*, we used the following filters:

```
(wlan.fc.type==1 && (wlan.fc.subtype==11 || wlan.fc.subtype==12)) || icmp
```

```
(wlan.fc.type==1 && (wlan.fc.subtype==11 || wlan.fc.subtype==12)) || tcp
```

This allowed us to filter RTS and CTS Wi-Fi packets and, in the first case, pings and in the second case the data transferred through TCP on *iPerf*.

In conclusion, there were some factors that made testing this topic way more difficult, such as:

- **Excessive amount of noise**. Even after applying filters, we would still get a lot of RTS and CTS packets that were not related to our tests, which made it very difficult to understand the relation between all packets captured. One solution could be going to a remote place, but that ended up not being feasible for any of us.
- **Monitor Mode**. Only late in the development of this project we managed to get the monitor mode working in the computer of one the team members (the others have Linux in a Virtual Machine (VM), which did not allow access to the network card natively), yet we didn't get a chance to assemble and work on the topic in more depth.
- **WiFi Jammer**. If we managed to get the WiFi Jammer working, and if it did what we expected, then we probably could have taken some conclusions about the behaviour of the control packets with more or less traffic.

Chapter 7

Conclusion

In conclusion, the process of instantiating and analyzing 802.11 network configurations was a valuable learning experience for our group. By examining the effect of various configurations, such as the use of the RTS/CTS mechanism, coverage and performance settings, roaming between APs and ad-hoc mode, we gained a better understanding about the *modus operandi* of a Wireless Local Area Network and the factors that can impact its performance and reliability.

Overall, this project demonstrated the importance of careful planning and configuration in the design and implementation of 802.11 networks and highlighted the potential of these networks to support a wide range of applications and use cases. Our results show that careful consideration of these configurations is crucial to optimizing a wireless network and getting the most out of it.

In future work, it would be interesting to continue to examine the performance of 802.11 networks in different environments and scenarios and further refine our understanding of the impact of other configurations on the Cisco Access Point.

To end this report, we would like to express our appreciation to our professors, Professor Daniel Corujo and Professor Francisco Fontes, for their guidance and support throughout the project. We would also like to thank senior technician António Alves, who provided valuable feedback and assistance on the equipment. Without the help of all these individuals, this project would not have been possible.

References

[1] Tonitrus. "Tonitrus: Cisco - AIR-CAP3602I-E-K9". (2021), [Online]. Available: https://www.tonitrus.com/es/redes/cisco/access-point-controller/cisco-3600-access-point/10106278-003-cisco-air-cap3602i-e-k9-802.11n-cap-w/cleanair-4x4-3ss-mod-int-ant-e-reg-domain/. (accessed: 22.12.2022).

[2] PuTTY. "Download PuTTY - a free SSH and telnet client for Windows". (), [Online]. Available: https://www.putty.org/. (accessed: 29.12.2022).

[3] Red Hat. "Linux tools: Getting the message out with dmesg | Enable Sysadmin". (), [Online]. Available: https://www.redhat.com/sysadmin/dmesg. (accessed: 29.12.2022).

[4] Cisco. "Chapter 1 - Using the Command-Line Interface". (), [Online]. Available: https://www.cisco.com/en/US/docs/wireless/access_point/12.4_3g_JA/command/reference/cr43cli_external_docbase_0900e4b180443050_4container_external_docbase_0900e4b180480623.html. (accessed: 29.12.2022).

[5] olgor.com. "WiFi Analyzer". (2022), [Online]. Available: https://play.google.com/store/apps/details?id=abdelrahman.wifianalyzerpro. (accessed: 19.12.2022).

[6] iPerf. "iPerf". (2016), [Online]. Available: https://iperf.fr/iperf-doc.php. (accessed: 18.12.2022).

[7] Microsoft. "Network Shell (Netsh) | Microsoft Learn". (2021), [Online]. Available: https://learn.microsoft.com/en-us/windows-server/networking/technologies/netsh/netsh. (accessed: 26.12.2022).

[8] Cisco. "Cisco Aironet 3600 Series Lightweight Access Points Getting Started Guide - Cisco". (2019), [Online]. Available: https://www.cisco.com/c/en/us/td/docs/wireless/access_point/3600/quick/guide/ap3600getstart.html#60734. (accessed: 26.12.2022).

[9] Intel. "Wi-Fi Roaming Aggressiveness Setting". (2021), [Online]. Available: https://www.intel.com/content/www/us/en/support/articles/000005546/wireless/legacy-intel-wireless-products.html. (accessed: 28.12.2022).

[10] R. Ramanathan and J. Redi, "A brief overview of ad hoc networks: Challenges and directions", *IEEE communications Magazine*, vol. 40, no. 5, pp. 20–22, 2002.

[11] Dan McInerney. "DanMcInerney/wifijammer: Continuously jam all wifi clients/routers". (2022), [Online]. Available: https://github.com/DanMcInerney/wifijammer. (accessed: 26.12.2022).

[12] Flerov (zyphex). "Start with Python2/3 result in str() errors · Issue #125 · DanMcInerney/wifijammer". (2022), [Online]. Available: https://github.com/DanMcInerney/wifijammer/issues/125. (accessed: 26.12.2022).