UNIVERSITAT POLITÈCNICA DE CATALUNYA

MASTER THESIS

High Speed S-band Communications System for Nanosatellites

Author:
MATIES PONS SUBIRANA

Supervisor:
Dr. ADRIANO CAMPS CARMONA

A thesis submitted in fulfillment of the requirements
for the degree of Master in Telecommunications Engineering

in the

Barcelona School of Telecommunications Engineering

June 22, 2016

UNIVERSITAT POLITÈCNICA DE CATALUNYA

Abstract

Barcelona School of Telecommunications Engineering
Department or School Name

Master in Telecommunications Engineering

High Speed S-band Communications System for Nanosatellites

by MATIES PONS SUBIRANA

In order to accomplish the objectives of the next generation of nanosatellites high-speed downlinks have to be designed. This goal faces stringent design constraints as nanosatellites are limit in power, processing capabilities and dimensions. In the quest for higher bit rates the widely used VHF band has to be replaced for higher frequency bands and the link budged margin tightened, decreasing the SNR at reception.

The proposed solution uses COTS 2.4 GHz WiFi adapters as transceivers. Range limitations imposed by the default 802.11 mode of operation are bypassed by using packet forging and injection at transmission jointly with monitor mode at reception. A loss-resilient unidirectional downlink is achieved by using application-layer encoding by means of LPDC-Staircase codes. This solution has been already implemented in ^{3}CAT-2, a 6 unit cubesat GNSS-R mission to be launched in July 2016. In addition, bursts of errors are combated by using Reed-Solomon.

The system has been tested under Doppler shift and scintillation effects, and a 188Km link between Barcelona and Mallorca has been performed, showing satisfactory results.

Acknowledgements

I owe my deepest gratitude to my thesis tutor Dr. Adriano Camps for giving me the possibility to become a member of the Nanosat Lab and providing me the opportunity to work on a field that I feel passionate about. His indefatigable spirit, enthusiasm and knowledge has motivated and inspired me throughout the project. He allowed this thesis to be my own work, but cleverly steered me in the right direction whenever he thought I needed it.

I would like to show my gratitude to Dr. Gregori Vazquez and Dr. Jaume Riba for his patience and expert advice. I greatly appreciate their feedback.

I am indebted to my Nanosat Lab colleagues for his support, guidance and ideas. I would like to remark the altruist aid I received from them during the testing stage.

I would like to specially thank my friend and classmate Santi for his invaluable advice and support. His help has saved me uncountable hours of work.

Last but not least, I must express my very profound gratitude to my family and beloved people for their unconditional support and persistent encourage. They have motivated me to pursue my dreams and they have always been on my side on all my endeavors. The completion of my master thesis wouldn't have been possible without them.

Contents

List of Figures

List of Tables

List of Abbreviations

ICBM	**I**nter**c**ontinental **B**allistic **M**issile
P-POD	**P**oly-**P**icosatellite **O**rbital **D**eployer
EPS	**E**lectrical **P**ower **S**ystem
LEO	**L**ow-**E**arth **O**rbit
VNIR	**V**isible (and) **N**ear **I**nfra**R**ed
GSD	**G**round **S**ample **D**istance
TLE	**T**wo-**L**ine **E**lement
NORAD	**N**orth **A**merican **A**erospace **D**efense **C**ommand
PLL	**P**hase-**L**ocked **L**oop
GNSS	**G**lobal **N**avigation **S**atellite **S**ystem
SNR	**S**ignal-(to)-**N**oise **R**atio
FCC	**F**ederal **C**ommunications **C**ommision
ISM	**I**ndustrial (,) **S**cientific (and) **M**edical
AP	**A**ccess **P**oint
LAN	**L**ocal **A**rea **N**etwork
RF	**R**adio **F**requency
COTS	**C**ommercial **O**ff-**t**he-**S**helf
NIC	**N**etwork **I**nterface **C**ard
WNIC	**W**ireless **N**etwork **I**nterface **C**ard
MAC	**M**edium **A**ccess **C**ontrol
HAL	**H**ardware **A**bstraction **L**ayer
SoC	**S**ystem **o**n **C**hip
PCB	**P**rinted **C**ircuit **B**oard
IR	**I**nfra **R**ed
DSSS	**D**irect **S**equence **S**pread **S**pectrum
FHSS	**F**requency **H**opping **S**pread **S**pectrum
DBPSK	**D**ifferential **B**inary **P**hase **S**hift **K**eying
DQPSK	**D**ifferential **Q**uadrature **P**hase **S**hift **K**eying
CCK	**C**omplementary **C**ode **K**eying
PLCP	**P**hysical **L**ayer **C**onvergence **P**rotocol
OFDM	**O**rthogonal **F**requency-**D**ivision **M**ultiplexing
FEC	**F**orward **E**rror **C**orrection
CTS	**C**lear **t**o **S**end
RTS	**R**eady **t**o **S**end
MIMO	**M**ultiple-**I**nput **M**ultiple-**O**utput
MCS	**M**odulation (and) **C**oding **S**cheme
GI	**G**uard **I**nterval
OSI	**O**pen **S**ystems **I**nterconnection
PMD	**P**hysical **M**edium **D**edepenent
CSMA/CA	**C**arrier **S**ense **M**ultiple **A**ccess **C**ollision **A**voidance
OS	**O**pertating **S**ystem
DS	**D**istribution **S**ystem
WEP	**W**ired **E**quivallent **P**rotocol

CCSDS	**C**onsultive **C**ommite (for) **S**pace **D**ata **S**ystems
3GPP	**3**rd **G**eneration **P**artnership **P**roject
MBMS	**M**ultimedia **B**roadcast (/) **M**ultimedia **S**ervice
DVB-H	**D**igital **V**ideo **B**roadcast (for) **H**andleld
IPDC	**I**nternet **P**rotocol **D**ata**C**asting
RSSI	**R**eceived **S**ignal **S**trength **I**ndicator
SDR	**S**oftware **D**efined **R**adio

Chapter 1

Introduction

This chapter presents a brief introduction to the Cubesat platform and explains the constrains associated to small satellites. It also provides an overview on the currently implemented high-speed downlinks and explains why WiFi devices are potentially good transceivers for nanosat high-speed communications.

1.1 History of Small Satellites and Cubesats

It is October 4, 1957, 10:29 p.m. Moscow time. The main engines of a R-7 Semyorka ICBM are fired at the Tyuratam launch base in the Kazakh Republic (nowadays Baikonur Cosmodrome). Minutes later the rocket delivers its main payload, the Sputnik satellite, into a LEO orbit. The URSS has inaugurated the space age [1].

The launch of the *Sputnik* satellite initiated the so-called space race in which the two most powerful countries in the world, the US and the URSS, competed to master the space exploration.

Fueled by the atmosphere of fear on the height of the Cold War, both nations invested large amounts of public money to take a leading position on this new battleground.

At the beginning the Soviets were ahead launching the first satellite and first human into space. Running behind the Soviet pace the US decided to concentrate their efforts on the ambitions plan to put a man on the Moon, which was accomplished in July 20, 1969 overtaking the URSS in the space race.

The space race supposed a giant advance in rocket technology leading to the construction of more reliable and powerful rockets. However, the access to space remained during several decades limited to governments or companies with large budgets.

It was not until the appearance of the Cubesat standard, proposed by Bob Twiggs and Jordi Puig-Suari in 1999, that the access to the space was democratized [2]. The authors proposed a launch interface between the launch vehicle and the small satellites called P-POD. A P-POD has enough room for accommodating 3 Cubesats of 10x10x11 cm in it. As the weight and volume of the P-POD is small it can be accommodated as secondary payload in the launch vehicle, taking advantage of a room that wouldn't otherwise be used.

The first Cubesat was launched in 2003 and in June, 2016 over 400 Cubesats had already been launched [3]. The popularity of this platform comes from the huge cost reduction, yet still providing sufficient space for developing practical satellites. With a production and launch cost that can be cut down to two-hundred thousand euros the Cubesats are more affordable for universities and small companies, that use them for educational, scientific or commercial purposes.

1.2 Cubesat Constrains

The small form factor of the Cubesats is a stringent limit that affects all the spacecraft subsystems.

1.2.1 Power

The electrical power on a Cubesat is managed by the Electrical Power System and is one of the most critical and scarce resources. To ensure that the power requirements of the subsystems are satisfied independently of the received solar radiation rechargeable Lithium-ion batteries are typically used. They are able to support peak power consumptions and to provide energy to the systems in eclipse conditions.

Solar panels are usually assembled to the external walls of the spacecraft, with a maximum effective area of $0.06\ m^2$ per Cubesat unit. This reduced solar cell surface area leads to low electrical power generation. Although deployable solar panels have been proposed and effectively used [4], [5] for increasing the cell's surface, their design is more complex, entails higher risks, and their use is not widespread.

The energy harvesting capabilities of the spacecraft sets an upper bound to the power consumption of the system. This limit affects all the subsystems that have to strictly follow it in order to comply with the power budged.

1.2.2 Computing

Onboard computers are in charge of controlling the payloads, storing data, keep logs from sensors, etc. Oriented to low-power consumption, the computers have usually reduced horsepower and CPU demanding tasks should be avoided in favor of highly optimized tasks.

Moreover, the selected microcontrollers and microprocessors along with the software used have to be fault-tolerant and robust against malfunctions.

1.2.3 Attitude Control

The attitude control system makes possible to maneuver the satellite in order to perform the desired tasks. For instance, when CubeSats are deployed, they receive an asymmetrical impulse that makes them tumble. Depending on the CubeSat mission they need to be stabilized by the attitude control system.

Earth observation missions may have scientific instruments with demanding pointing requirements. The attitude control can also be used to maximize the solar irradiance on the solar cells.

To determine the satellite's orientation and maneuver it, special equipment is used as reaction wheels, gyroscopes, Sun sensors, Earth sensors, reaction wheels, magnetorquers, etc. They are usually combined to decrease the attitude control and determination error in a wide range of possible scenarios. Due to major advances in miniaturization, the attitude control systems of Cubesats are getting closer in performance to the ones implemented in bigger satellites.

1.2.4 Communications

In order to establish communications with Earth and to send data the CubeSats have to rely on wireless telecommunications. Due to the stringent limits in power and the reduced physical dimensions that difficult boarding high-gain antennas, the links implemented in CubeSats are mostly used for telemetry and low volume data download as they achieve low-to-moderate bit rates. Ground stations are equipped with high-gain antennas and powerful amplifiers to ensure sufficient received power margin.

The VHF amateur band of 433 MHz is the most used radio band for non-commercial CubeSats [6]. However, future missions require links of higher bit rates and high-speed downlinks are being studied.

1.3 High-Speed Downlinks Survey

The inherent limitations of energy, processing power and size combined with the particularities of satellite channel makes the design of high-speed downlinks a challenging task. It is not possible to use high power amplifiers without compromising the power budged, the limited physical size constrains the design of large directive antennas and the usually reduced processing power of the onboard processor establishes an upper bound on the communications algorithms that may be used.

Due to that limitations the number of CubeSats that feature a high-speed downlink (>1 Mbps) is very low. The following table shows some examples of already launched CubeSats with downlink data rates equal or higher than 1 Mbps:

TABLE 1.1: High-Speed Downlinks implemented in CubeSats, data obtained from [7].

CubeSat	Frequency	Maximum Bit Rate
Hayato	13.275 GHz	1 Mbps
DICE	465 MHz	1.5 Mbps
CINEMA	2.2 GHz	1 Mbps
DOVE	8.2 GHz	4 Mbps

While the number of already deployed satellites that feature high-speed downlinks is scarce, several studies and prototypes have been developed for future use. The next table is a survey of some of them:

Board	Frequency	Maximum Bit Rate
Be-1/Be-2	2/4 GHz	2 Mbps
SRI	4/6 GHz	5 Mbps
Indigo	3.1/3.4 GHz	5 Mbps
CANOPUS	25.5/27 GHz	40 Mbps
Syrlinks	8/8.4 GHz	50 Mbps

TABLE 1.2: High-speed prototyped transceivers. Data from [8], [9],[10], [11] and [12] respectively.

As the table shows, the proposed solutions move towards higher frequency bands including the S-band, C-band, X-Band and Ka-Band achieving maximum theoretical throughput of several tens of Mbps. However, these transceivers are manufactured from scratch and their production cost and complexity is high. In fact, as a rule of thumb the higher is the RF frequency, the higher is the cost. In addition, some of this prototypes work under licensed frequency bands, so the cost of paying the permissions to work under licensed frequencies has also to be taken into account.

1.4 Use of open ISM bands for CubeSats downlinks

ISM bands are unlicensed frequency bands intended to be used by industrial, scientific and medical apparatus. However, these bands may also be used by communication equipment that has to tolerate interference coming from ISM devices.

The 2.4 GHz ISM band that spans from 2400 to 2483.5 MHz is nowadays worldwide available. The world-coverage availability makes this band very interesting for satellite links as communications may be established anywhere disregarding borders.

A myriad of connectivity products employ this band due to the tax-free use of the spectrum. Several standards working on the 2.4 GHz ISM band have been developed the past two decades becoming some of them of widespread use as Bluetooth or WiFi.

The following table summarizes the most common standards operating in the 2.4 band and the interesting characteristics from the point of view of long range links:

TABLE 1.3: Comparison of standards that use the 2.4 GHz ISM band.

Technology	Range	Power	Bit Rate
Bluetooth	10 m	Low	1 Mbps
LoRa	15 Km	High	0.3/50 kbps
ZigBee	100 m	Low	250 kbps
WiFi	300 m	High	1/300 Mbps

Although the LoRa standard promises long operational ranges its inherent low bit rate makes this technology non well suited for high-speed links. On the other hand, WiFi may achieve several hundreds of bits per second although its coverage is usually moderate.

1.5 Advantages of using WiFi products

After almost two decades since the first 802.11 products were launched, nowadays WiFi products are ubiquitous and mature. They can be found in a large variety portable devices ranging from laptops to cameras and also providing connectivity to dozens of apparatus like smart TV's, radios, etc.

Driven by the demands of the market, advanced communication techniques have been incorporated in the last standards. With over 40 active WiFi chipset vendors [13] the economy of scale makes 802.11 compliant devices inexpensive and of high quality.

As a result, WiFi devices can be used as inexpensive radio transceivers that are capable of providing several Mbps links in a license-free band. These three main features make this technology potentially well-suited to be used for high speed satellite communications.

1.6 Conclusions

Chapter 1 has briefly explained the history of Cubesats indicating that the popularity of this spacecrafts comes from the reduced manufacture and launch costs.

It has been seen how the reduced size of Cubesats entails constraints that effect all the systems. Particularly, the communications are affected in terms of output power and computing costs of signal processing algorithms. The high-speed downlink survey has shown that nowadays the number of high-speed downlinks implemented in Cubesats is scarce. For obtaining higher throughput traditional VHF links are being replaced by links operating at S-Band, X-Band or Ka-Band.

The chapter has also explained that there are communications standards using the license-free ISM bands which offer high bit rates as the 802.11-family standard. With proper modifications, WiFi standard may be used for satellite communications.

Chapter 2

^{3}CAT-3 Data Downloading Requirements

^{3}CAT-3/MOTS is a remote sensing 6U cubesat sponsored by the Institut Cartogràfic de Catalunya and the Institut d'Estudis Catalans to be launched in 2019. Its main scientific payloads are a multispectral camera and an L-Band radiometer.

This chapter describes the orbital parameters of the ^{3}CAT-3 mission and explains the downloading requirements of the main payload. The grounds stations that will be used and the link budged are also discussed, paying special attention to the scintillation effect that will degrade the signal. Several link budget improvements are proposed at the end of the chapter.

2.1 LEO Polar Sun-Synchronous Orbit

^{3}CAT-3 will be a satellite designed to perform Earth observation from orbit. The Sun illumination will play an important role on the scientific data gathering as the multispectral camera has to work in day-light conditions. Most Earth observations satellites are placed in polar sun-synchronous orbits in order to monitor the Earth surface with constant sunlight conditions and be able to cover the whole Earth in a small number of orbits.

From the point of view of the Sun, during one day the Earth rotates a little bit more than 360 degrees (approximately 361 degrees). That is because while rotating, the Earth is also orbiting the Sun. As a result, in one year the orbit plane spins 360 degrees. Because of that, a perfect polar orbit of 90 degrees of inclination wouldn't see the same illumination when passing over the same ground track on different days.

Polar Sun-Synchronous orbits have their nodal regression rates equal to the rate in which the Earth travels around the sun [14]. To accomplish that, Sun-synchronous orbits take advantage of the the Earth geopotential irregularities. More specifically, the oblateness J_2 term of Earth geopotential is used to find combinations of very specific altitude and inclinations in which the perturbation that causes the nodal regression counteracts the approximately extra degree of rotation per day, obtaining a Sun-synchronous orbit.

The following figure shows an example of a Sun-synchronous orbit at three consecutive orbits:

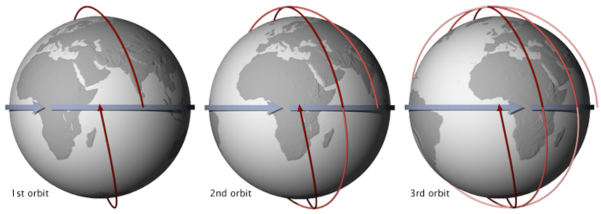


FIGURE 2.1: Sun-Synchronous orbit, taken from [15].

It can be appreciated how the orbit is not perfectly polar as it has more than 90 degrees of inclination. The figure also shows that the satellite crosses the equator with the Sun illuminating the ground track always with the same conditions.

Remote sensing satellites that perform science missions are usually placed in the Sun-synchronous orbit because it maintains the surface's sunlight illumination as consistent as possible. However, there are still small changes in the illumination angle due to the effect of the Earth inclination. The illumination consistency makes possible to compare images from the same season with others of any year. As the illumination is maintained approximately constant, there are not significant differences in shadows and lightning that could create illusions of change. With the Sun-synchronous orbit is easier to track changes over time and obtain consistent data [15].

2.2 Satellite-Ground Station distance

The following figure describes the geometry of the scenario, a spherical Earth and circular orbit has been considered:

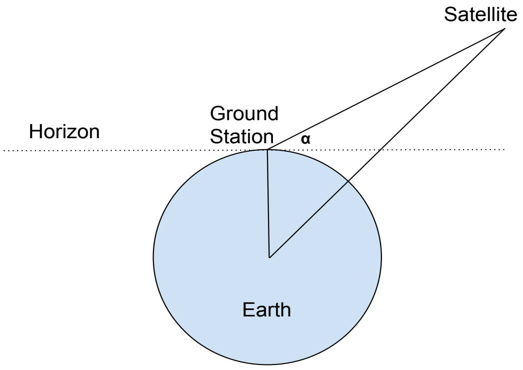


FIGURE 2.2: Link distance scheme.

The satellite orbits the Earth at a particular distance from the surface that is the orbit's height. The ground station is placed at the Earth surface, and the distance between the ground station and the satellite as a function of the elevation angle α may be calculated as follows:

From the resulting triangle two sides and one angle are known. Applying the cosine law we have that:

$$c^2 = a^2 + b^2 - 2 \cdot a \cdot b \cdot cos(\gamma), \tag{1}$$

being c the distance between the Earth center and the satellite. Equivalently it may be expressed as: $c = R_{Earth} + Orbit_{Heigth}$.

The variable a is the earth radius and the angle α the elevation angle above the horizon plus 90 degrees. Thus, $\gamma = \alpha + 90$.

In order to obtain the distance as a function of the elevation angle the equation is rearranged and the squares completed:

$$a^2 - 2 \cdot a \cdot b \cdot cos(\gamma) + b^2 - c^2 = 0, \tag{2}$$

$$a^2 - 2 \cdot a \cdot b \cdot cos(\gamma) + b^2 - c^2 = 0, \tag{3}$$

$$b^2 - 2 \cdot a \cdot b \cdot cos(\gamma) = -a^2 + c^2, \tag{4}$$

$$b^2 - 2 \cdot a \cdot b \cdot cos(\gamma) + a^2 \cdot cos^2(\gamma) = a^2 \cdot cos^2(\gamma) - a^2 + c^2, \tag{5}$$

$$(b - a \cdot cos(\gamma))^2 = a^2 cos^2(\gamma) - a^2 + c^2, \tag{6}$$

$$b - a \cdot cos(\gamma) = \pm\sqrt{a^2 \cdot cos^2(\gamma) - a^2 + c^2}, \tag{7}$$

$$b = a \cdot cos(\gamma) \pm\sqrt{a^2 \cdot cos^2(\gamma) - a^2 + c^2}. \tag{8}$$

Substituting by names:

$$Distance = R_E \cdot cos(\gamma) \pm\sqrt{(R_E)^2 \cdot cos^2(\gamma) - (R_E)^2 + (R_E + Orbit_H)^2} \tag{9}$$

The distance will be the positive solution of this equation.

The United Nations approved in 2008 a guideline [16] to limit the lifetime of spacecrafts to 25 years in order to mitigate space debris. As cubesats don't have decommissioning equipment, the maximum allowed orbit would be of around 600 km, depending on the ballistic coefficient.

Taking into account that value, the following figure shows the distance between the ground station and the satellite as a function of the elevation angle:

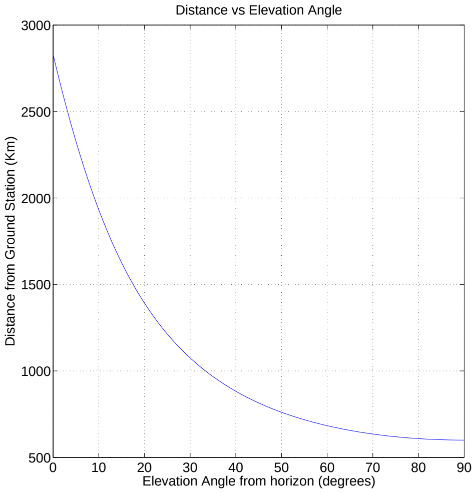


FIGURE 2.3: Distance ground station to satellite as a function of the elevation angle for a 600 km orbit.

Usually the orography that surrounds the ground stations (buildings, trees, mountains, etc) block the horizon and the link is usually established for elevation angles above 10 degrees. Taking into account this assumption the range of distances spans from 1930Km at elevation angle 10 degrees to 600 km when the elevation angle is 90 degrees.

2.3 Downloading requirements

The main objective of ^{3}CAT-3 is to obtain multispectral images of the Earth and especially from the Catalonia region. The photos will be taken at 4 or 5 different bands including: VNIR: 440-510 nm, 520-590 nm, 620-680 nm, 690-730 nm and 850-890 nm. The expected swath of the images will be between 30 and 50 km with a GSD up to 5 m [17].

2.3.1 Data budget

Featuring a high resolution multispectral camera at 5 bands the amount of gathered data will be huge.

The following figure shows the multispectral data acquisition process:

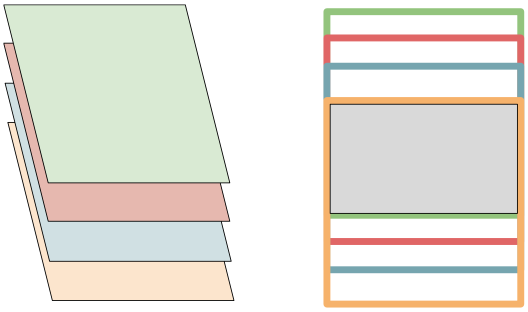


FIGURE 2.4: Multispectral data acquisition.

As the camera is not able to take a photo and save the data of all bands at the same time it takes stills for each band consecutively. While the photos are being captured the satellite is moving at 7.5 km/s and consequently the photos do not perfectly overlap. To solve this problem the onboard computer has to determine which region of the photos is common for all bands, crop the images and obtain the remaining multispectral valuable data.

Each image without compression is expected to have a size around 70 MB. A multispectral image composed by 5 of them would be around 350 MB.

2.3.2 Contact Time and Transmission Speed

The contact time between the satellite and ground station will highly depend on the orbit parameters and where the grounds stations are placed. For instance, the Svalbard ground station is able to track almost all the polar passes whereas the UPC ground station will be only able to track some passes. The contact time for each visible pass will vary from 10 to 15 minutes depending on the maximum elevation angle [18]. For a satellite pass of 10 minutes the minimum average throughput to download 350 MB of data should be:

$$throughput_{min} = \frac{350 \cdot 8}{10 \cdot 60} = 4.7\,\mathrm{Mbps}. \tag{10}$$

2.3.3 Doppler effect

The Doppler effect produces a frequency shift of the electromagnetic waves when there is a relative movement between transmitter and receiver. When the distance between the emitter and the receiver decreases the frequency increases, whereas if the distance increases the frequency decreases.

Satellites placed in LEO orbits are particularly exposed to large Doppler effects due to the low altitude of their orbit, and higher velocities.

The speed of an orbiting object placed on a radius r is given by:

$$v \approx \sqrt{\frac{G \cdot M}{r}}. \tag{11}$$

The approximation comes from considering a perfectly circular orbit and the mass of the satellite negligible as compared to the mass of the orbited object (i.e the Earth). For the ^{3}CAT-3's expected orbit of 600 km and with $G = 6.67 \times 10^{-11}\,\mathrm{m^3\,kg^{-1}\,s^{-1}}$ and $M = 5.97 \times 10^{24}\,\mathrm{kg}$ the speed would be:

$$v \approx \sqrt{\frac{G \cdot M}{r}} = \sqrt{\frac{6.67 \cdot 10^{-11} \cdot 5.97 \cdot 10^{24}}{600 \cdot 10^3 + 6370 \cdot 10^3}} = 7.56\,\mathrm{km/s}, \tag{12}$$

And the Doppler shift that an emitted frequency f_0 experiences as a function of the velocity is expressed as:

$$f = \frac{c + v_r}{c + v_s} \cdot f_0, \tag{13}$$

where v_r is the speed of the receiver, and v_s the speed of the satellite. Considering that the ground station is static with respect to the satellite the maximum Doppler shift that the communications will suffer at 2.45 GHz will be:

$$\begin{aligned}\Delta_f &= \pm\left(2.45\cdot 10^9 - \frac{3\cdot 10^8+0}{3\cdot 10^8+7.56\cdot 10^3}\cdot 2.45\cdot 10^9\right)\\ &= \pm 61.738\,\mathrm{kHz}\end{aligned} \tag{14}$$

This maximum value would correspond to a nadir pass of the satellite when the relative speed between the satellite and the ground station is maximum. Taking into account that a WiFi channel is 22 MHz of bandwidth wide this shift in frequency causes a fractional displacement of:

$$FractionalShift = \left(\frac{61.73\cdot 10^3}{22\cdot 10^6}\right)\cdot 100 = 0.28\% \tag{15}$$

2.4 UPC and Svalbard Ground Stations

Ground stations are in charge of establishing the communications from Earth to cubesats and vice-versa. They are a key element of the mission, as all the commands and data are send and received wirelessly.

Ground stations are designed to ensure reliable communications and they have the ability to track satellites.

The ^{3}CAT-3 mission will have two operative ground stations. The main one will be located at the UPC in Barcelona and the second in Svalbard, near the North pole.

2.4.1 UPC Ground Station

The ground station located at the D3 building's roof has been installed to be used on the ^{3}CAT-2 mission and will be reused for the ^{3}CAT-3.

Purchased from ISIS [19] as a full ground station kit solution, the ground station features a 3 meter dish mesh antenna with 35.4 dBi of gain [20]. The antenna comes with a tracking system with 0.5 degrees of resolution with an available upgrade to 0.1 degrees. The dish antenna is mounted on the top of a 6 meter post to elevate its position and avoid surrounding obstructions.

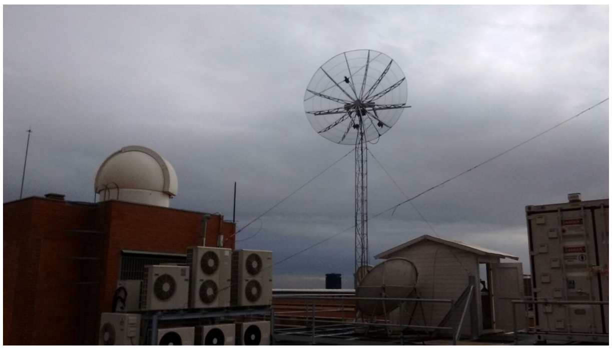


FIGURE 2.5: UPC Ground Station.

In order to gather more data and improve the contact time with the satellite a ground station located near the North pole will be rented.

2.4.2 Svalbard Ground Station

Polar Sun-synchronous orbits pass near the North pole every 90-100 minutes. For that reason the poles are excellent locations to place ground stations and maximize the contact time per day.

Svalbard is a Norwegian archipelago situated between 74 and 81 degrees of latitude and with a population of 2600 inhabitants is one of the northernmost human settlements in the world [21]. Since 2004 the archipelago is connected with Norway with a redundant fiber optic cable.

Taking advantage of all these characteristics that make Svalbard a privileged site to place ground stations the SvalSat company was born. Nowadays it is a commercial ground station that provides connectivity products to polar orbits in the C, L, S, X, and Ka-band.

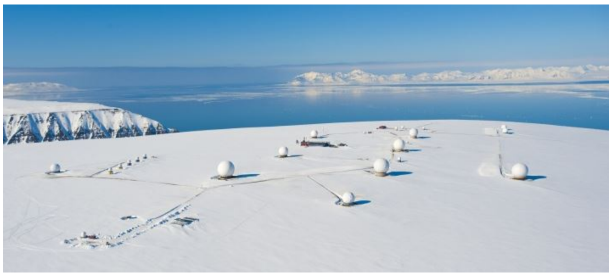


FIGURE 2.6: Svalbard Ground Station, taken from [21].

For the ^{3}CAT-2 mission this ground station facilities will be rented to communicate with the satellite at least two times per day. For ^{3}CAT-3 they will be probably rented again. This will mean a huge increment on the contact average time per day being it possible to download much more data.

2.5 Link Budget

When calculating the link budged all the elements involved on the link have to be taken into account. In this section the details of each one will be described.

2.5.1 Ideal Received Power

If the emitted power of the antenna would be radiated isotropically the power intensity would be:

$$U(\theta,\phi) = \frac{P_T}{4\pi} \text{ [W/sr]}, \tag{16}$$

but using directive antennas the radiation is not the same in all directions, so we express the gain as a function of the direction:

$$G(\theta,\phi) = \frac{U(\theta,\phi)}{P_T/4\pi}. \tag{17}$$

The maximum gain is the direction in which the antenna is radiating more power density:

$$G_{MAX} = \frac{U_{MAX}}{P_T/4\pi}. \tag{18}$$

The gain of an antenna is expressed as compared to the gain of the isotropic case where $U_{MAX} = 1$:

$$G(dBi) = 10log(G_{MAX}). \tag{19}$$

Then, the power density can be calculated from the Equivalent Isotropic Radiated Power (EIRP):

$$EIRP(\theta,\phi) = P_T \cdot G(\theta,\phi), \tag{20}$$

and the power density (flux) can be expressed as:

$$\Phi(\theta,\phi) = \frac{P_T \cdot G(\theta,\phi)}{4\pi d^2} = \frac{EIRP(\theta,\phi)}{4\pi d^2}. \tag{21}$$

In order to collect this energy a receiving antenna is used. Assuming a parabolic dish with an area of:

$$Area_R = \pi R^2 = \pi \frac{D^2}{4}, \tag{22}$$

and taking into account the effective area of the antenna ($Area_{Reff} = \eta \cdot Area_R$) the captured power can be expressed as:

$$P_R = \Phi \cdot Area_R. \tag{23}$$

Due to reciprocity theorem [22], the gain of the receiving antenna can also be expressed as:

$$G_R = \frac{4\pi}{\lambda^2} \cdot A_{Reff}, \tag{24}$$

then the received power is:

$$P_R = \frac{EIRP \cdot A_{Reff}}{4\pi d^2} = \frac{EIRP \cdot G_R}{(\frac{4\pi d}{\lambda})^2}. \tag{25}$$

The above formula expresses the received power for a simplified model where no losses of any kind other than the distance are taken into account.

2.5.2 Free Space Losses

The free space losses take into account the loss of signal strength of an electromagnetic wave that would result from a line-of-sight path through free space, with no obstacles to cause reflection or diffraction.

$$Losses_{FreeSpace} = \left(\frac{4\pi d}{\lambda}\right)^2 \tag{26}$$

2.5.3 Pointing Losses

Obtaining a perfect alignment in both antennas is virtually impossible, so we have to compute the looses associated to this effect. Supposing that the pointing accuracy will be good and so $\theta_{deviation} \leq \Delta\theta_{3dB}$, being $\Delta\theta_{-3dB}$ the half power beamwidth:

$$G \approx G_{MAX} - 12\left(\frac{\theta_{deviation}}{\Delta\theta_{3dB}}\right)^2, \tag{27}$$

so the pointing losses associated will be:

$$L_{PointingRX} \approx 12 \cdot \left(\frac{\theta_{deviation}}{\Delta\theta_{3dB}}\right)^2. \tag{28}$$

Being $\Delta\theta_{3dB} \approx 70 \cdot \left(\frac{\lambda}{D}\right)\ degrees$, where D is the diameter of the antenna.

The previous two formulas say that the more directive is an antenna the bigger will be its pointing loss for the same pointing error.

Taking into account that the ground station will have much higher gain than the satellite antenna, the pointing losses associated with the ground station will be much larger than the ones associated to the satellite's antenna. Moreover, as the ^{3}CAT-3 main payload would be a multispectral imagery system and will have a high-precision attitude system with low $\theta_{deviation}$, the pointing losses associated to the satellite antenna are negligible.

The parabolic antenna of 3 meters of diameter placed in the UPC has a $\Delta\theta_{3dB}$ of:

$$\theta_{3dB} \approx 70 \cdot \left(\frac{\frac{3\cdot 10^8}{2.45\cdot 10^9}}{3}\right) = 2.86^{\text{o}}. \tag{29}$$

The total pointing losses will depend on two main factors:

- NORAD TLE
- Antenna Pointing Capabilities

The satellite's orbital parameters are usually known prior to launch with a large margin of error. Once the satellite is in orbit it is possible to track its position using radars. The NORAD is an organization from the United States that tracks all detectable objects in Earth's orbit and makes the orbital parameters of non-confidential objects publicly available [23] in the TLE form.

The accuracy of the TLE for LEO nanosatellites is usually below 1 km, although higher error values in the order of 10 km have been reported for low elevation LEO orbits ($\approx 400km$) were the atmospheric drag is considerably higher [24]. As the ^{3}CAT-3 is expected to be launched in a $\approx 600km$ orbit, the expected error of the NORAD TLE is below 1 km. This error assumes only a maximum deviation angle of 0.072 degrees. However, if for some reason the NORAD updates are not coming as regularly as they usually are, this error will increase.

Once the position of the satellite is known, the antenna tracking system has to point the antenna to the satellite. The encoders of the tracking system have a step of 0.5 degrees, for that reason, under optimal conditions, the maximum pointing error would be of ± 0.25 degrees. However, there are more elements that will increase the pointing error. Although the antenna reflector is a mesh and offers low resistance to the wind, moderate breezes will certainly induce vibrations to the structure. The tracking system may also provoke some shaking because of the stepped motors.

The next table shows the two main pointing error contributors. A margin of 1 degree has been applied to not underestimate other possible causes of error.

Cause	Error (degrees)
NORAD TLE	0.072
Tracking System	0.25
Margin	1
Total	1.32

TABLE 2.1: Causes of pointing error and margins applied.

The pointing losses supposing 1.32 degrees of pointing error would be:

$$L_{PointingRX} \approx 12 \cdot \left(\frac{1.32}{2.857}\right)^2 = 2\,56dB \tag{30}$$

The Svalbard ground station presumably has much better tracking system and consequently lower pointing error than the UPC ground station.

The following figure shows the associated pointing losses as a function of the deviation angle in degrees:

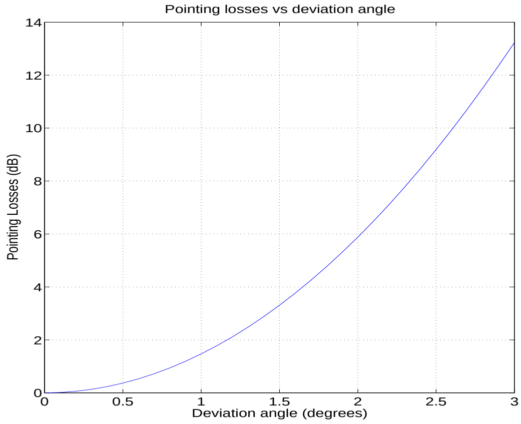


FIGURE 2.7: Pointing losses vs deviation angle for UPC ground station.

As it can be seen, it follows a parabolic curve so it is very important to maintain the deviation angle as low as possible.

2.5.4 Atmospheric effects

The lossy effects induced by the atmosphere can be decomposed between tropospheric effects and ionospheric effects.

Troposheric effects

The tropospheric effects include gas absorption and rain attenuation. Two gases are the main absorption contributors: Oxygen and water vapor. The following graph shows this two contributions separately as a function of the frequency, being the red curve oxygen, the blue curve water vapor, and there is painted in black the sum of both effects.

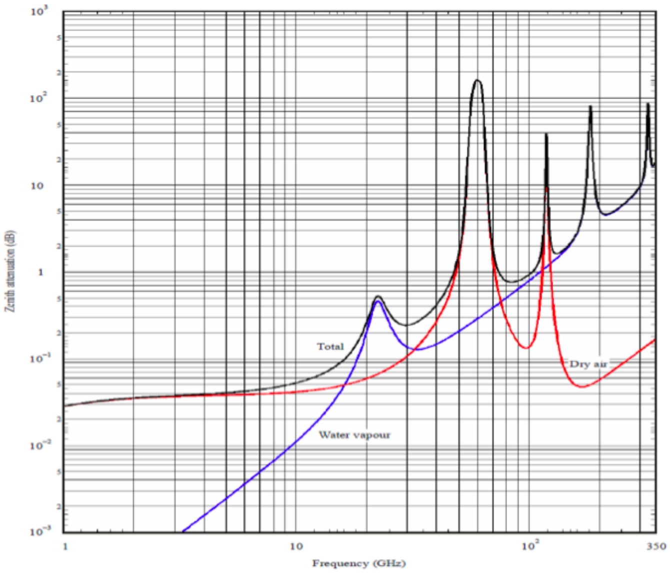


FIGURE 2.8: O2 and H20 vapor attenuation vs frequency, taken from [25].

As the plot shows, there are two absorption peaks at 22 GHz for water vapor or 60 GHz for oxygen. However, for frequencies below 10 GHz the contribution of the atmospheric absorption is negligible.

Ionospheric effects

The highest layer of the atmosphere is called the ionosphere and contains particles that are ionized by the Sun's radiation. Layers of electrons are created usually following a stratified structure, but diverse factors may lead to anomalies in form of clouds of electrons that create ionospheric disturbances [26].

Radio signals get distorted when passing through the ionosphere. From the communications point of view the major problems come from polarization rotations and scintillation effects. On the one hand, the polarization rotation (Faraday rotation) can be combated by using circular polarized antennas. On the other hand, scintillation effects introduce rapid phase changes and deep fading that present a major handicap to overcome and consequently need to be studied in detail.

The presence of irregular electron densities at the ionosphere that cause scintillation effects has been studied and quantified. The following graph provides a global view on how frequent are the ionospheric disturbances.

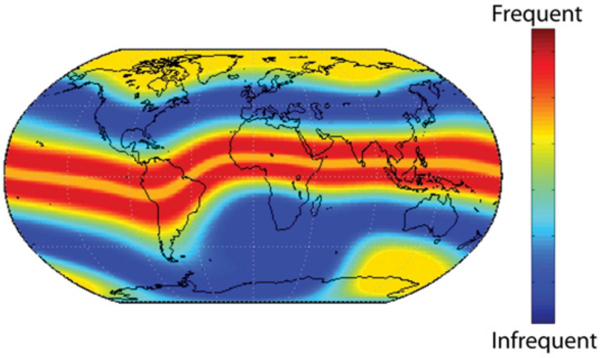


FIGURE 2.9: Frequency of ionospheric disturbances, taken from [27].

As the image shows, there are two distinguishable lines in regions within ±20 degrees from the geomagnetic equator where the scintillation effects are more frequent. It is also noticeable that polar zones are also prone to suffer from these effects. Taking into account that data, the Svalbard ground station situated near the North pole is more likely to suffer from scintillation effects than the UPC receiving facilities, placed in a mid-latitude.

Scintillation effects that change amplitude and phase are often described with two parameters: S_4 and σ_ϕ.

The S_4 index is defined as the standard deviation of the average of the normalized signal intensity. The index reflects the change in the intensity of the signal amplitude [28]:

$$S_4 = \sqrt{\frac{\langle S_I^2 \rangle - \langle S_I \rangle^2}{\langle S_I \rangle^2}}, \tag{31}$$

where $\langle S_I \rangle$ is the mean of the signal intensity. Values of S_4 below 0.3 are considered as low scintillation and usually do not cause problems in communications systems. However, due to the stringent constraints of the nanosat platform the designed link will presumably work under almost no margin. In these conditions even small scintillation effects associated with low S_4 indexes may have a remarkable impact on the system, diminishing the received signal power under the critical threshold and creating outage.

The other parameter used to describe the scintillation effects is σ_ϕ, and it is defined as the standard deviation of the carrier phase in radians:

$$\sigma_\phi = std(\varphi), \tag{32}$$

where φ is the carrier phase. Large fluctuations on the carrier phase may lead the PLL work out of their specifications and provoke errors. This is a serious problem for GNSS systems that operate at L-Band as GPS outages are critical to many applications [29]–[31]. Fortunately, the scintillation effects are have less impact as the frequency increases and the following relations apply:

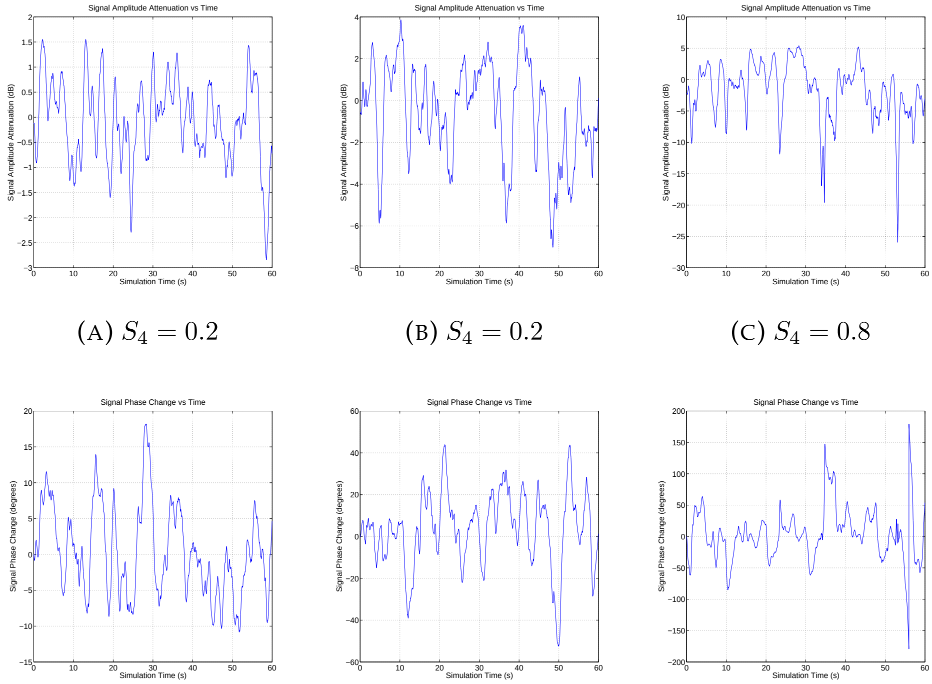
$$S_4 \propto \frac{1}{f^{1.5}} \tag{33}$$

$$\sigma_\phi \propto \frac{1}{f} \tag{34}$$

For that reason, the attenuation in amplitude intensity and the phase variation at S-Band will be considerably lower than in L-band.

The used model accepts as inputs the already explained S_4 and also τ_0. This new variable expresses the decorrelation time of the complex fading process. It basically describes how quickly the phase and magnitude change, so it is intrinsically related with σ_ϕ. The lower is that value the more quickly the phase and magnitude will change. At extreme conditions τ_0 is below 0.2 s, whereas in moderate and calm conditions this value is around 1 s and 2 s respectively [32].

The following figures show the scintillation effects at the 2.4 GHz frequency for different values of S_4 and $\tau_0 = 1$ s. The top row of images shows the intensity changes in amplitude that the signal will suffer. The main problem associated with that fluctuations will be the deep fadings that will cause loss-of-signal. The bottom row images show the phase changes for the same channel realizations.



(A) $S_4 = 0.2$

(B) $S_4 = 0.2$

(C) $S_4 = 0.8$

(D) $S_4 = 0.2$

(E) $S_4 = 0.5$

(F) $S_4 = 0.8$

FIGURE 2.10: Intensity and phase changes for S_4 0.2, 0.5, 0.8 values with $\tau_0 = 1$ s.

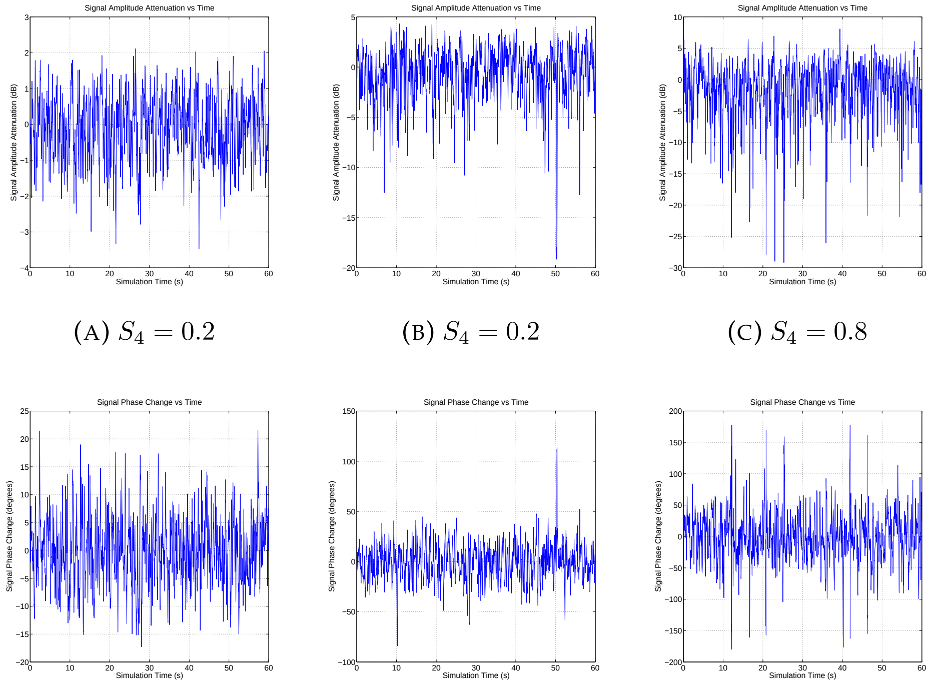
The first figure represents a low value of S_4. This would be a value that could be expected at the Barcelona ground station. Usually the peaks are between ±1 dB although there are some that surpass the 2 dB attenuation. In principle, that scintillation effect shouldn't have a strong impact in the link's quality albeit they can slightly degrade it, and produce sporadic loss-of-signal.

The B figure represents a moderate value of the S_4 parameter to be expected in the Svalbard ground station or at Barcelona in worst-case conditions. With this value of S_4 fadings of 4 dB are usual and sporadic deep fades of +10 dB too. That scintillations effects could compromise the link quality and provoke outage periods.

The C image shows the scintillation effects for a 0.8 S_4 index. That value could be seen at Svalbard during geomagnetic storms. In that case, fadings of 10 dB are usual and peaks of +20 dB of attenuation may occur. This would completely interrupt the link and a large quantity of packets would be lost.

The phase changes are plotted in images D, E and F. It can be appreciated that the higher is the S_4 index the more steep are the phase changes. For $S_4 = 0.2$ the phase changes of ±20 degrees are usual. This value increases to ±30 and ±50 for S_4 indexes 0.5 and 0.8 respectively.

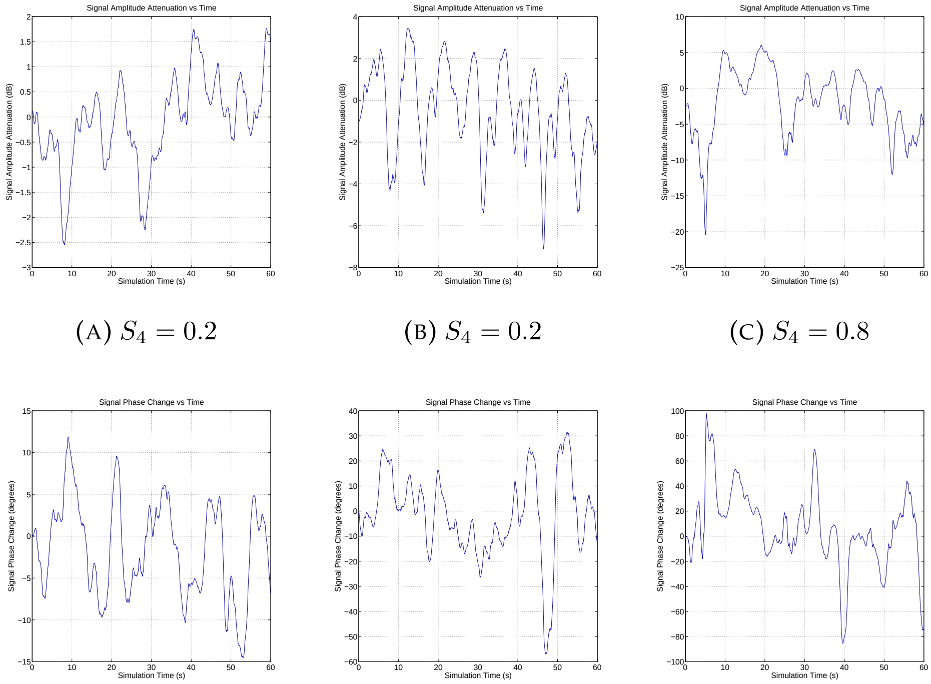
Steep phase changes will stress PLL's and may create loss-of-lock. The steepness of the phase changes is dependent of the τ_0 value. The former 6 images were plotted with $\tau_0 = 1$ s, the following figures show the intensity and phase changes for the same values of S_4 but with τ_0 values of 0.1 s and 2 s:



(A) $S_4 = 0.2$ (B) $S_4 = 0.2$ (C) $S_4 = 0.8$

(D) $S_4 = 0.2$ (E) $S_4 = 0.5$ (F) $S_4 = 0.8$

FIGURE 2.11: Intensity and phase changes for S_4 0.2, 0.5, 0.8 values with $\tau_0 = 0.1$ s.



(A) $S_4 = 0.2$ (B) $S_4 = 0.2$ (C) $S_4 = 0.8$

(D) $S_4 = 0.2$ (E) $S_4 = 0.5$ (F) $S_4 = 0.8$

FIGURE 2.12: Intensity and phase changes for S_4 0.2, 0.5, 0.8 values with $\tau_0 = 2$ s.

It can be seen that for $\tau_0 = 0.1$ s the changes are much more steep.

From the point of view of the intensity this could be a problem if the duration of the packets was large enough for one packet suffer several fadings. This way, almost no packet would arrive uncorrupted as erasures inside the packets would be frequent. However, at high data rates the packet duration is really low, so it shouldn't be a problem. From the phase changes point of view, smaller τ_0 will stress the PLL much more and the odds of loss-of-lock will be higher. For $\tau_0 = 2$ s the changes are smoother. On the one hand, this will imply that deep fadings will have longer duration and the outage lapses will be higher. On the other hand the PLL shouldn't have much problems to track the phase changes.

2.5.5 SNR Link Budged Approach

An approach for testing communication links is by means of computing the ratio between signal and noise at the reception chain. For a certain desired bit error rate the demodulator will need a minimum SNR to work with, and thus, the link characteristics can be obtained.

The SNR at the demodulation will depend on how much signal has been received and how much noise has been introduced.

Thermal Noise

Any material that is at any temperature higher than 0 K has electrons that are in movement independently of the applied voltage. This motion of charges creates a noise whose power spectral density is nearly constant throughout all the frequencies. This noise is called white noise and it has the following noise power:

$$P_{ThermalNoise} = k \cdot T \cdot B \tag{35}$$

Where k is the Boltzmann constant: 1.39×10^{-23} J K^{-1}, T is the physical temperature, and B is the bandwidth.

Any communications reception chain will suffer from the effects of the thermal noise. For SNR-critical deep-space missions cryogenic refrigeration with liquid helium is used to lower the noise power as much as possible [33]. However, this cryogenic system are really expensive and non-affordable for the university. The expected thermal noise power for the ground station at 290 K will be of -204 dBW/Hz. For 22 MHz 802.11b channel the power of the thermal noise is around -100.6 dBm.

The demodulation requirements of WiFi adapters in terms of CNR or SNR are not usually specified in the devices' datasheets difficulting the link budged calculation using this approach.

2.5.6 Initial Link Budged

The datasheets of WiFi adapters specify their sensitivity at different data rates. As the primary objective of the link budged is to determine the how much power is received and compare this value with the adapter's specifications there is no need to compute the CNR of the received signal and then compare it with the final E_b/N_0 at the demodulator, it is just necessary to determine how much power is received.

The received power is given by the following formula:

$$P_R = P_T + G_T - L_T - L_{\theta_{TX}} - L_{PathLoss} - L_{Atm} - L_{\theta_{RX}} + G_R - L_R - L_{Pol} \quad (36)$$

The next table summarizes the elements of the link budged calculation for typical values that may be achieved:

TABLE 2.2: Initial Link budged.

Feature	Value	Calculation
P_T	2W/33 dBm	
G_T	7 dBi	
L_T	0.5 dB	
$L_{\theta_{TX}}$	0 dB	
$L_{PathLoss}$	1930/600 km	165.75 dB/155.6 dB
L_{Atm}	0 dB	
L_{RX}	2.56 dB	
G_R	3 m, 55%	34.95 dB
L_R	0.5 dB	
L_{Pol}	1 dB	
Received Power		-95.36/-85.21 dBm

The maximum distance has been taken according to the calculations explained in 2.2.

It is also worth noting that this table omits the scintillation effects, so they have to be later taken into account over the average received power.

WiFi adapters have typically a minimum receive sensitivity of -95 dBm at the minimum data rate of 1 Mbps and around -90 dBm while working at 6 Mbps. The obtained results indicate that the link margin is almost null when operating at maximum distance at 1 Mbps and the 5 Mbps downlink target wouldn't be meet in these conditions for the whole range of distances.

However, with around 5 dB more of received power the scenario would be completely different and high speed communications could be established during all the satellite pass-by.

In order to grasp this extra dBs of power the elements that influence on the link budged will be studied.

2.5.7 Link Budged Improvements

From the components of the link budged formula G_T, G_R and P_T are usually the link design variables that have the strongest impact in the received power.

The orbit altitude has been set at 600 km to calculate the worst-case condition from the point of view of the link budged. However, it is probable that the final orbit will have 500 or 400 km of apogee, supposing an increase in the received power of 1.6 dB and 3.5 dB, respectively.

Transmitted Power

Electrical power in a cubesat is a scarce resource that needs to be thoughtfully allocated. For that reason, all the subsystems are designed to operate with minimum power consumption.

Communication systems have usually intermittent and peaky consumption because they are turned on when there is visibility with the ground station. To satisfy the power budged the communications subsystem should consume as minimum as possible. Having 2 W of output power using class A-B amplifiers means a consumption of around 6 W. A slight improvement in the effective output power could be attained with a combination more efficient amplifiers and modulations that are robust to non-linearities. For instance, using 802.11b modulations less linear amplifiers are required.

Low Noise Amplifier

In order to determine how much noise is introduced in the receiving chain, the noise factor is used:

$$F = \frac{SNR_{in}}{SNR_{out}} = \frac{SNR_{in}}{\frac{G \cdot S_{in}}{G \cdot (N_{in} + N_{device})}} = 1 + \frac{N_{device}}{N_{in}}, \tag{37}$$

Using the noise factor any device can be modeled as an additional source of noise by means of $T_{device} = (F - 1) \cdot T_0$.

A typical receiving chain can be decomposed in the antenna, the cable and connectors, and the receiver block. The receiver block has a LNA, mixers, filters, etc. The noise figure of cables coincides with the loss they introduce. The equivalent temperature of the overall system may be expressed as:

$$T_{sytem} = T_1 + \frac{T_2}{G_1} + \frac{T_3}{G_1 G_2} + \cdots + \frac{T_n}{G_1 \cdots G_{n-1}}. \tag{38}$$

The first element in the expression is the antenna temperature $T_1 = T_{ant}$. The antenna picks up noise from celestial and also from terrestrial by means of possible backlobes or spillover. This temperature can range from a few Kelvin to thousands if pointed to the sun. The second element could represent the LNA and the following ones all the rest of the receiving chain.

It is important to note that the LNA noise figure and gain will play an important role on the overall value of the T_{system} when placed at first instance, as G_1, the gain of the low noise amplifier, will be multiplying the denominator of the following terms. For that reason, a cheap and straightforward way of improving the reception chain is to place a LNA with a good figure of merit at the beginning of the receiving chain.

Typical noise figures for receiving chains of WiFi devices are around 5 dB. The receiving chain can be greatly improved by placing a very low noise amplifier in front of the receiving chain. For the 2.4 GHz band there LNAs with noise figures around 0.6 dB and 10-15 dB of gain. Placing one of these amplifiers would drastically enhance the overall system noise, as it will lower the noise figure of the system around 4 dB.

Ground Station

The larger the effective area of the ground station, the higher the received power. The UPC ground station is a considerably large 3 meters parabolic dish antenna with approximately 35 dBi of gain. A larger antenna would be much more expensive and would have demanding pointing requirements. However, if the pointing losses are reduced from 1.3 degrees to 0.5 degrees, which could be feasible, a gain of 2 dB in the received power would be obtained. As this pointing accuracy would require of a really fine adjustment will not be considered as a potential improvement.

Satellite Antenna Gain

Due to the reduced dimensions of the cubesat platform there is not much room for the antennas. Patch antennas have been used for the 2.4 GHz band typically achieving gains of 6-8 dBi [34], [35].

From the elements of the link the satellite antenna is the one with more potential to offer several dB's more of gain without affecting the rest of the mission.

Deployable antennas have been recently started to be designed with the pursue of increasing directivity while maintaining the compactness required to fit inside the cubesat structure. Although some deployable designs are engineering masterpieces that involve complicated mechanism [36], more simple antenna designs as Yagi-Udas, log-periodics, helicals or hemispherical-helicals antennas [37] have been proposed and tested with promising results. These antennas are able to achieve gains in excess of 12 dBi, that suppose 5 dB of gain over a patch antenna.

2.5.8 Improved Link Budged

The following link budged takes only into account the improvement in the transmitting antenna while keeping the other parameters unmodified.

TABLE 2.3: Link budged with improvements.

Feature	Value	Calculation
P_T	2 W/33 dBm	
G_T	12 dBi	
L_T	0.5 dB	
$L_{\theta_{TX}}$	0 dB	
$L_{PathLoss}$	1930/600 km	165.75 dB/155.6 dB
L_{Atm}	0 dB	
L_{RX}	2.56 dB	
G_R	3 m, 55%	34.95 dB
L_R	0.5 dB	
L_{Pol}	1 dB	
Received Power		-90.36/-79.71 dBm

With these range of reception power values and the use of an external LNA to lower the noise figure of receiver it would be possible to maintain a +6 Mbps link using off-the-shelf WiFi adapters during all the satellite pass-by, effectively downloading the entire 350 MB of imagery data.

Nonetheless, at low elevation angles the link margin would be close to null while working at maximum speed. The solution would be to use lower bit rates at low elevation angles, increasing the link's margin at the expense of decreasing the throughput.

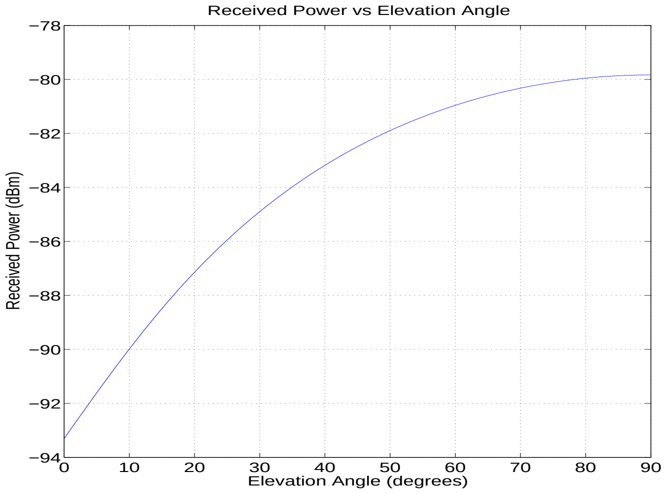


FIGURE 2.13: Received power as a function of the elevation angle.

2.6 Conclusions

Chapter 2 has described the characteristics of the polar Sun-synchronous orbit that ^{3}CAT-2 and ^{3}CAT-3 missions will have. It has been seen that the link distance will vary between 2000 km, for 10^o of elevation above ground, and 600 km , in case of a zenith pass.

The chapter has explained the downloading requirements of the ^{3}CAT-3, whose multispectral camera will take photos of around 70 MB per frequency band. In order to download a multispectral photography during a 10 minute satellite pass a bit rate of around 5 Mbps is needed. The maximum Doppler shift of the WiFi carrier frequency will be ± 61.74 kHz.

The characteristics of the UPC ground station have been discussed and the increase in gathered data when renting the Svalbard ground station has been explained.

The link budged calculation has shown that using a usual setup the received power is insufficient to ensure high-speed communications during all the fly-by. However, if the satellite has moderate gain antennas (12 dBi) and in reception a LNA is used, the average received power is enough to maintain 5 Mbps during the satellite pass. Even in that case, the scintillation effect will produce loss-of-signal that has to be combated.

Chapter 3

Use of WiFi

This chapter describes the implementation of a robust S-Band communication system, revealing the hardware and software used as well as the modifications that have been done to adapt COTS hardware to the characteristics of the satellite link.

The chapter starts with a brief history of WiFi to explain more in depth the layers 1 and 2 implemented in 802.11 devices. Then, the incompatibilities of the WiFi standards for long range operation and the previous studies in this topic are explained. After that, the custom proposed solution and the hardware selection are described.

3.1 History of WiFi

The invention of radio communications systems back in the beginning of the XX century was a major breakthrough that had a profound impact in many industries. The untethered long range communications were born and quickly got the attention of the military sector, that saw in this new technology a must-have perk for the battlefield.

The history says that for every new weapon or technology that has been created, giving to the owner a temporal superiority over the enemy, a countermeasure system has been developed to defeat it. The same happened with radio communications, and the first radio jammers appeared. These devices were used to block the narrowband communications by injecting noise at the target frequency. Radio communications were no longer reliable and systems using them could be potentially compromised and put out of order.

The solution to this thread came from Hedy Lamarr a Hollywood actress who patented in 1942 a Frequency Hopping Spread Spectrum system. That system avoided the jammers' interference by spreading the signal over rapidly changing frequencies.

The use of spread-spectrum technology remained restricted to military uses until 1985 when the FCC decided to allow the use of spread spectrum over three bands: 900 MHz, 2.4 GHz and 5.8 GHz. These bands were already allocated for ISM applications, and thus the new devices working on this frequency band had to deal with interference. The new devices using this band would use Direct Sequence Spread Spectrum modulation to ensure interoperability in congested environments due to its relative immunity to interference.

In 1988 the 802.11 committee was created to conduct the standardization process to ensure interoperability between devices of different manufacturers. In 1997 the first 802.11 standard was released, providing speeds of 1

Mbps and 2 Mbps. In 1999 two new standards were released, the 802.11b provided a speed improvement and the 802.11a incorporated the 5 GHz band as well as providing more bit rate [38].

The same year the WiFi alliance was formed to certify the compatibility of a device to the 802.11 standards. The branding campaign of this certifying body became effective and nowadays most of the people refer to the 802.11 technology as WiFi.

Since then, the deployment of high-speed broadband Internet access to the home and the proliferation of laptop and portable devices has boosted the use of WiFi products. The standardization process has continued to meet the market requirements, in terms of bandwith, delay, coverage, etc. New modulations as the OFDM with higher order constellations are now used to provide higher throughputs (introduced in 802.11a), as well as 40 MHz channels and MIMO (introduced in 802.11n) or advanced techniques as beamforming and up to 160 MHz channel bandwidth (introduced in 802.11ac).

Nowadays WiFi continues to evolve to cover the broad spectrum of demands. New promising bands as the 60 GHz one, for supporting multiple-gigabit data rates for room-range connectivity, or the sub 1 GHz band, for long range communications, are being studied [39].

3.2 Physical Layer

The OSI model describes the physical layer as the lowest layer which is in charge of the bit-transmissions through the medium. In case of the 802.11 the physical layer is subdivided in two sublayers namely: PMD and PLCP.

The PMD or physical medium dependent layer is in charge of the modulation and demodulation of the frame. The PLCP or Physical Layer Convergence Protocol creates the physical frame to be send, including the PLPC preambles and headers.

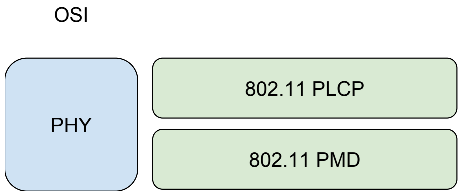


FIGURE 3.1: OSI Physical layer and 802.11 equivalent sublayers.

The physical layers of the 802.11 standards are always implemented in silicon to achieve real time operation without interrupts. In order to introduce more advanced communications techniques the physical layer has been changing.

3.2.1 802.11-legacy

The 802.11 standard was launched in 1997 and three different physical layers were specified:

- Diffuse Infrared operating at 1 Mbps.
- Frequency Hopping Spread Spectrum operating at 1 or 2 Mbps.
- Direct Sequence Spread Spectrum operating at 1 or 2 Mbps.

Manufacturers started to develop hardware following the specifications. Although the IR physical layer was defined in the standard it was never implemented and DSSS was preferred over FHSS.

This legacy implementation was rapidly supplemented by the standard 802.11b released in 1999. Nowadays, only very obsolete hardware operate with the 802.11-1997 physical layer so the description of its physical layers will be omitted.

3.2.2 802.11b

The popularization of WiFi technology came in 1999 when the 802.11b standard was released and Apple introduced WiFi connectivity to their iBook laptop. The new standard offered 1, 2 and 11 Mbps bit rates and was backwards compatible with the 802.11-legacy standard.

Channelization

The ISM band at 2.4 GHz spanning from 2400 MHz to 2484 MHz was split in channels of 22 MHz bandwidth with 5 MHz channel spacing. This frequency allocation created up to 14 communication channels, however, due to local regulations on the spectrum not all the channels could be used worldwide.

The following table contains for each channel its central frequency and area of application:

TABLE 3.1: WiFi channels defined in 802.11b.

Channel Number	Central Frequency	Area
1	2412	Europe, US, Japan
2	2417	Europe, US, Japan
3	2422	Europe, US, Japan
⋮	⋮	⋮
11	2462	Europe, US, Japan
12	2467	Europe, Japan
13	2472	Europe, Japan
14	2484	Japan

In order to maintain the backward compatibility the new standards have maintained the same channels in the 2.4 GHz band.

As the channels have a bandwidth of 22 MHz and they are spaced by only 5 MHz there can only be 3 non-overlapping channels (or 4 in case of Japan when using channel 14). A common combination is to place the AP's at channels 1, 6 and 11. The following image exemplifies that configuration:

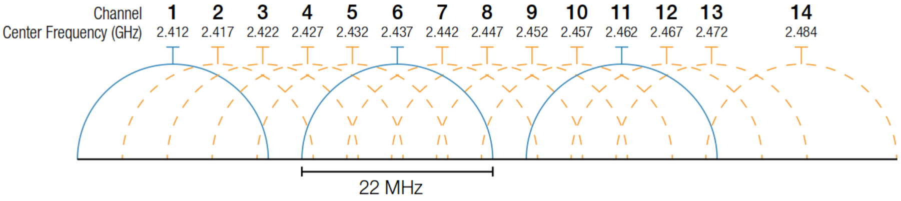


FIGURE 3.2: WiFi channel distribution, taken from [40].

The 802.11b standard describes a spectral mask to be applied on each channel to assure that the power leakage into other channels is under control. The next figure shows how the mask has to be applied:

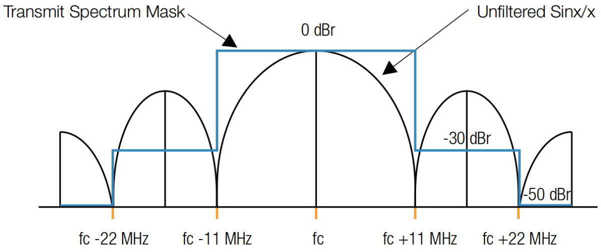


FIGURE 3.3: 802.11b spectral mask, taken from [40]

The figure illustrates the absolute value of a $sinc$ function in frequency coming from a square signal of 11 MHz (because the chip rate is 11 Mchips/s). As the picture shows, the peak of the first side lobe has to be 30 dB below the power of emitted in the central frequency and the second side lobe peak below 50 dB.

Modulation

The 802.11b standard defines 4 bit rates achieved by means of different modulations.

TABLE 3.2: 802.11b Modulations and Bit rates.

Modulation	Bit Rate
DBPSK	1 Mbps
DQPSK	2 Mbps
CCK	5.5 Mbps
CCK	11 Mbps

To obtain the first two raw bit rates a Barker code of 11-chip is used at 11 Mchips/s. The purpose of Barker codes is to detect and correct errors in transmission and help the receiver to get synchronization. The Barker code of length 11 is applied to every bit spreading the signal by a factor of 10.

The chips resulting from the application of the Barker code are modulated with DBPSK and DQPSK for obtaining 1 and 2 Mbps.

DBPSK and DQPSK were defined in the 802.11-legacy standard. In order to demodulate BPSK and QPSK a synchronized local oscillator is needed to recover the phase. During this process it may be phase ambiguity that may lead to an incorrect phase lock and consequently the recovered bits are inverted.

In order to solve this problem differential codification can be used. In that case the information is mapped to the modulator taking into account the current and past symbol. The general concept of differential codification is to convey the differences between symbols.

The price to pay for removing the phase ambiguity is that when the receiver makes an incorrect decision two consecutive symbols will be incorrect, because of the codification memory [41].

The phase changes applied when encoding with DBPSK and DQPSK are summarized in the following table:

TABLE 3.3: Differential encoding for BSPK and QBPSK, from [42].

DQPSK	Phase Shift	DBPSK	Phase Shift
00	0	0	0
01	$\pi/2$	1	π
11	pi		
10	$3\pi/2$		

The 802.11b standard introduced a new way to increase the maximum throughput by means of using a CCK code. In that case the 11-chip Barker code is substituted by the 8-chip CCK code. The 11 Mchips/s chip rate is the same in order to maintain the spreading properties and channelization. To obtain the 5.5 Mbps data rate 4 data bits are mapped to a 8-chip sequence. The 11 Mbps data rate is accomplished by mapping 8 data bits to the 8-chip sequence.

Frame Structure

In order to make possible the interoperability between devices that operate at different data rates the 802.11b standard establishes a frame format that has a preamble and a header that are usually send at 1 Mbps to ensure maximum robustness. The following figure shows how is this frame format specified:

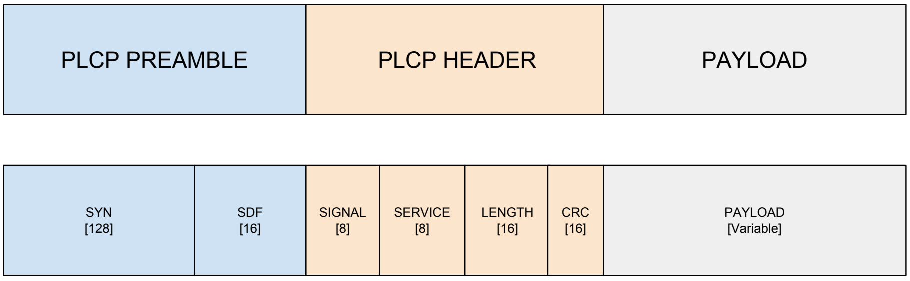


FIGURE 3.4: 802.11b physical frame.

The PLCP preamble contains two fields. The SYN consists of 128 bits that are used for gain setting, energy detection, antenna selection and frequency offset compensation. The SFD field has a length of 16 bits and its used for bit synchronization.

The PLCP Header contains 4 fields. The SIGNAL field of 8 bits contains information about the bit rate in which is modulated the rest of the PLCP header. The SERVICE field is reserved for future uses. 16 bits of LENGTH indicate which is the total size of the packet and finally a CRC of 16 bits protects the SIGNAL, SERVICE and LENGTH fields.

The standard also describes a short preamble that halves the transmission time of the PLCP preamble and header. That reduction is accomplished by decreasing to 64 the SYN field bits and transmitting the PLCP header at 2 Mbps instead of 1 Mbps.

The payload may be of variable length as indicated and contains information coming from higher layers.

3.2.3 802.11a/g

Pursuing higher data rates, a change of paradigm in modulation schemes was proposed for 802.11a and 802.11g standards, moving from the spread spectrum approach to OFDM modulations.

Although the 802.11a standard was released in 1997, as the 802.11-legacy standard, it took some years from the manufacturer to launch 802.11a compliant devices. This new standard was working at the 5 GHz ISM band and had a maximum raw throughput of 54 Mbps.

However, when the 802.11a products started to be available on the market the use of 802.11b was widespread. In addition, the first generation of 802.11a devices performed poor and were more expensive than 802.11b products [40].

The 802.11g standard was released in 2003 and copied the physical layer of the 802.11a to achieve up to 54 Mbps at the 2.4 GHz band of the spectrum.

As mentioned before, in order to provide backward compatibility with former standards the channelization remained the same.

Modulation

The abandon of the spread spectrum techniques in favor of OFDM was leveraged by the desire of obtaining higher data rates. In fact, it is difficult to achieve high bit rates using spread spectrum due to the large bandwidth and processing needed [43].

The principle of OFDM is to use multiple orthogonal subcarriers to transmit information. This way, a high speed stream is converted into several parallel low-speed transmissions. With the addition of the cyclic prefix, OFDM systems are able combat multipath-interference.

The OFDM modulation scheme proposed in 802.11a/g uses a 16.25 MHz wide channel divided into 52 subcarriers spaced 312.5 kHz from which 48 are used to transmit data and 4 serve as pilot signals.

The subcarriers are modulated using BPSK, QPSK, 16-QAM or 64-QAM and they are protected with convolutional FEC codes of rate 1/2, 2/3 or 3/4.

The standard describes the following combination of modulations and FEC rates:

TABLE 3.4: 802.11a/g combinations of modulations and FEC coding rates.

Modulation	Coding Rate	Bit Rate
BPSK	1/2	6 Mbps
BPSK	3/4	9 Mbps
QPSK	1/2	12 Mbps
QPSK	3/4	18 Mbps
16-QAM	1/2	24 Mbps
16-QAM	3/4	36 Mbps
64-QAM	2/3	48 Mbps
64-QAM	3/4	54 Mbps

Frame Structure

The 802.11g standard has several frame structures defined. The following one is the most extended one [44]:

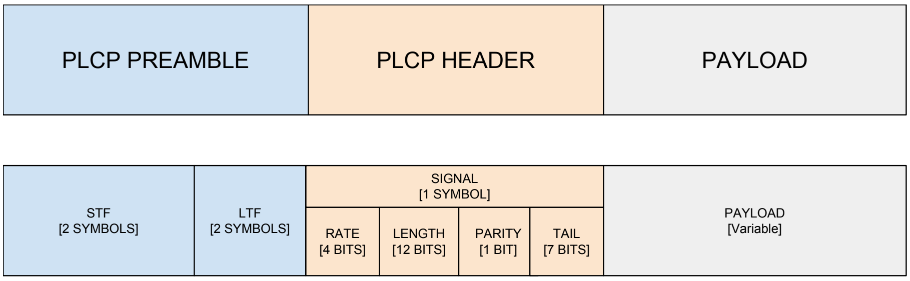


FIGURE 3.5: 802.11g physical frame.

The preamble has two fields. The STF stands for short training sequence and is used for initial timing synchronization and frequency estimate. It has the particularity of using only one fourth of the subcarriers. The next field is the Long Training Field and is used to perform fine training and frequency synchronization, as well as to estimate the channel response.

The PLCP header has one field called Signal, which is sent using BPSK with FEC 1/2 that is subdivided in 4 parts. The first one indicates the data FEC coding and modulation, that is, the 8 possibilities that the 802.11g standard has. The length field says how many bytes are in the payload. The following field is a 1-bit parity check of the rate and length fields. Finally the tail field is used to encode the Signal symbol with a FEC of rate 1/2.

The payload data is sent using 48 data subcarriers that may be modulated with BPSK, QPSK, 16 QAM or 64 QAM modulation and 4 pilot subcarriers used to monitor and track the frequency, phase and amplitude variations of the received data during the burst.

However, this frame structure is not compatible with the former standards because they are not able to understand the OFDM symbols. For that reason, two main strategies were developed to guarantee interoperability between 802.11b and 802.11g devices. The first one is to use the CTS and RTS mechanism, that is explained in 3.3.1. This way, even if the other devices are not able to understand the preamble collisions will be avoided. The second one consists of appending to the 802.11g frames a 802.11b PLCP preamble and header at the beginning.

The following figure shows how the first and second solutions are implemented highlighting how the addition of the DSSS preamble introduces a notable overhead (not in scale) in comparison with the OFDM preamble. For that reason, the price to pay for having interoperability between 802.11g and 802.11b standards its an overall reduction of throughput.

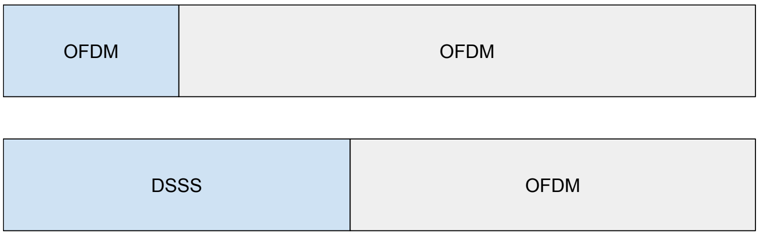


FIGURE 3.6: 802.11g Interoperability solutions.

3.2.4 802.11n

In 2009 the standard 802.11n was issued with the main purpose of providing higher bit rates over the network that the ones were delivering 802.11b and 802.11g.

The maximum theoretical raw data rate was increased form the previous 54 Mbps maximum to 600 Mbps. This one-fold increase was accomplished by combining several techniques: MIMO, larger bandwidth and higher modulation-orders. Moreover, it was added the possibility to operate in the 2.4 GHz and 5 GHz ISM bands.

Channelization

The use of larger bandwidth changed the channelization that had been constant since the 802.11-legacy standard. 802.11n can work with the former 20 MHz channelization or use 40 MHz channels.

The problem of operating 40 MHz channels in the 2.4 GHz band is that only 1 or 2 channels fit depending on regional regulations. Thus, the 40 MHz channelization is mostly used in the 5 GHz band where more bandwidth is available.

However, there is a lot of equipment in the market that supports only 802.11n in the 2.4 GHz band.

Modulation

The following table summarizes in a compressed format the main modulations and coding schemes standardized in the 802.11n standard for 20 MHz channels and long GI:

TABLE 3.5: 802.11n modulation and coding schemes.

Index	Stream	MCS index	MOD	FEC	RATE
0	1-4	$index + (stream - 1) \cdot 8$	BPSK	1/2	$6.5 \cdot stream$
1	1-4	$index + (stream - 1) \cdot 8$	QPSK	1/2	$13 \cdot stream$
2	1-4	$index + (stream - 1) \cdot 8$	QPSK	3/4	$19.5 \cdot stream$
3	1-4	$index + (stream - 1) \cdot 8$	16-QAM	1/2	$26 \cdot stream$
4	1-4	$index + (stream - 1) \cdot 8$	16-QAM	3/4	$39 \cdot stream$
5	1-4	$index + (stream - 1) \cdot 8$	64-QAM	2/3	$52 \cdot stream$
6	1-4	$index + (stream - 1) \cdot 8$	64-QAM	3/4	$58.5 \cdot stream$
7	1-4	$index + (stream - 1) \cdot 8$	64-QAM	5/6	$65 \cdot stream$

As the table shows, there are 8 different combinations of modulation and FEC rate. For each of this combinations there are four more combinations depending on how many spatial streams are being used. That gives a total of 32 bit rates, that range from 6.5 Mbps to 260 Mbps.

In case of using 40 MHz channels the data rates are roughly doubled. The standard also have the possibility to use a short GI in environments with short delay spread. This increases the data rate by a 10.7%. Using 4 spatial streams, short GI and a 40 MHz channel the bit rate may reach 600 Mbps.

Is it worth mentioning that the 802.11n standard uses 56 subcarriers from which 4 are used as pilots to monitor the phase/frequency and amplitude changes of the channel. This compares to the former 52 carriers (48 data + 4 pilots) of the 802.11g standard.

Frame Structure

802.11n has three defined frame structures: Legacy Mode, Mixed Mode and Greenfield Mode.

In legacy mode the 802.11n devices use the frame structure of the 802.11g standard. It is a backward compatibility mode that emulates 802.11g devices, and all the benefits introduced in 802.11n are lost.

The mixed mode appends the same PLCP preamble and header used in 802.11g to each one of the 802.11n packets. That way, 802.11g devices are able to correctly interoperate with the new standard, as they can understand the preamble and detect transmissions and 802.11n devices can benefit from the improvements of the new protocol.

Finally, the standard also defines a frame structure to be used in 802.11n-only networks. This frame structure is called Greenfield and this mode is not compatible with previous standards. The application of this new frame structure has performance benefits, although the use of it in networks were not all the devices are not 802.11n-compliant will decrease the overall throughput.

The frame structure of the Greenfield mode is depicted in the following image:

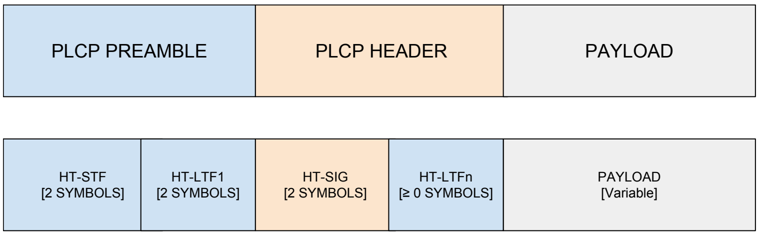


FIGURE 3.7: 802.11n Greenfield frame.

The PLCP preamble has two main fields, the HT-STF stands for high-throughput short training field and is used for the same functions as the STF of 802.11g although it is not compatible with it because of the 52 data subcarriers used. Similarly, the HT-LTF is used for the same objectives as the LTF of the 802.11g standard.

The PLCP header has the field HT-SIG that carries information required interpret the HT packet format [45]. After this field, there is an optional field that corresponds to the PLCP preamble called HT-LTFs and is used for MIMO training.

In order to make the 802.11n protocol to be backward compatible with 802.11g devices the following two solutions are used:

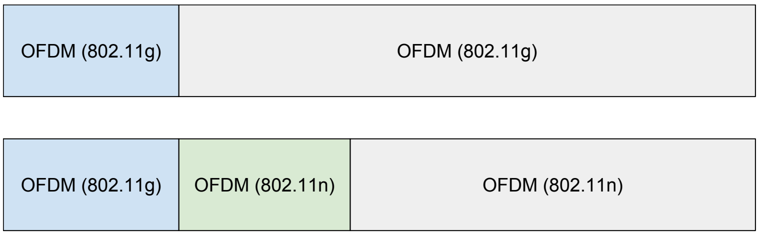


FIGURE 3.8: 802.11n Interoperability frame solutions.

The first solution called legacy mode is basically emulating a 802.11g frame, in which the preamble and payload are modulated with the OFDM established in 802.11g-compliant products.

The second solution adds a 802.11g preamble to a 802.11n frame. This solution is called mixed mode.

3.3 Link Layer

The link layer described in the OSI model is responsible to transfer data between adjacent nodes and also to detect and correct the errors that may happen in the physical layer. It is worth noting that the independence between the OSI layers is maintained and while different physical layers have been standardized the link layer has remained constant.

The functionalities of the OSI link layer are separated in two sublayers in the 802.11 scheme as the following figure shows:

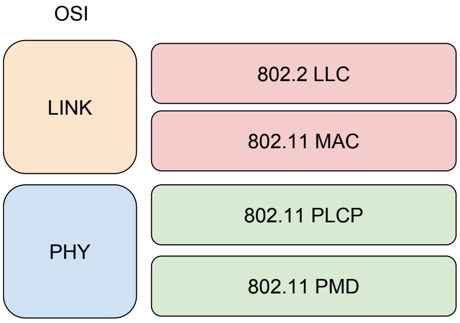


FIGURE 3.9: OSI Link layer and 802.11 equivalent sublayers.

The 802.11 MAC sublayer is in charge of coordinating the access to the medium and to use protocols for improving communications.

The 802.2 LLC sublayer, as the name indicates, it is not a part of the 802.11 protocol. It is used to provide a single data link control for all IEEE standardized LANs. The 802.2 LLC makes the MAC sublayer transparent and simplifies the interoperatibilty between LANs that implement different MACs.

3.3.1 802.11 Medium Access Control layer

The radio spectrum is a limited resource that WiFi devices have to share. The mechanisms that manage the allocation of the medium are placed in the 802.11 MAC layer.

Carrier Sense Multiple Access/Collision Avoidance

The principle of Carrier Sense Multiple Access is to sense the medium that needs to be shared to know whether the medium is being used or not. If the medium is being used the transmitter will hold, in order to avoid a collision.

An additional problem of wireless systems is that the radio transceivers are not able to listen while transmitting. For that reason, if two radios are transmitting simultaneously none of them will notice the interference created by the other and so, the collisions will last during the entire transmission.

Collision Avoidance is the solution to this problem. This way, devices not only apply CSMA but also try to avoid collisions. The CSMA/CA mechanism is based on the following structure:

1. Sense the medium.
   (a) If the channel is free for some amount of time, then transmit.
   (b) If channel is busy, wait until the transmission has finished.
      i. When the transmission has finished, wait a bounded random time (backoff). If after this random time the channel is free, transmit. If it continues being busy, wait a bounded random time smaller than the previous wait.

The CSMA/CA mechanism also tries to maintain fairness between devices. It gives priority to devices that have been waiting for a long time. The CSMA/CA protocol implements fairness between devices by means of the random backoff algorithm, which is bounded and decreases each time the transmission is hold. As summarized before, when a station detects the medium busy it waits a bounded random time. If the random waiting time wasn't applied, several stations willing to transmit data would eventually sense the medium free and would try to transmit data exactly at the same instant, creating multiple collisions.

Acknowledgements

To improve efficiency only positive ACK's are send. If no ACK is received after having send a packet and waited a certain time, the packet is considered lost.

Fragmentation

Wireless systems have bit error rates that may be in the order of $10^{-5} - 10^{-6}$. For long packets containing more than 10^{5} bits the probability of transmitting with errors a large packet could approach 1.

Fragmenting large packets into smaller ones solves this problem and is also implemented in the 802.11 MAC layer, which has a maximum packet size of 1500 Bytes.

Hidden Node Problem - Ready to send/Clear to Send

The hidden node problem is represented by the following figure:

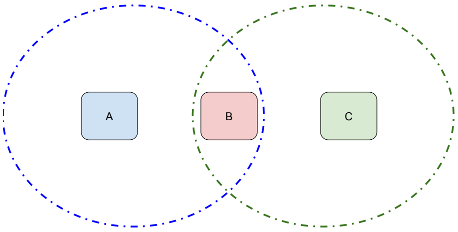


FIGURE 3.10: Hidden Node Problem.

In the figure 3.10 there are three 802.11 devices. The node B acts as AP and stations A and B transmit data to it. The circles represent the coverage of the signals transmitted by devices A and C. In that case, the terminals A and C wouldn't notice the presence of each other, and the CSMA/CA mechanism will fail.

To solve this problem the RTS/CTS mechanism is used. Basically, it consists of a way of controlling the access to the medium by means of short messages that are send by the devices to indicate the willingness to transmit data and by the AP to permit medium access.

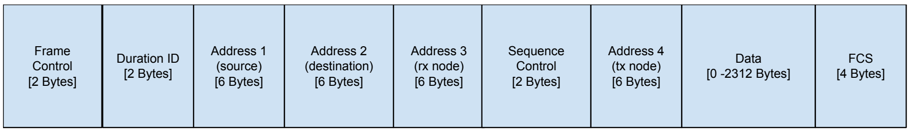
Node Identification

To identify each node in the network each device has a unique MAC address. This MAC address is hardcoded on the physical device and it has a length of 6 Bytes. The first 3 Bytes identify the manufacturer and the least 3 the card itself.

When the devices interchange messages the MAC address is contained in the packet.

Medium Access Control Frame Format

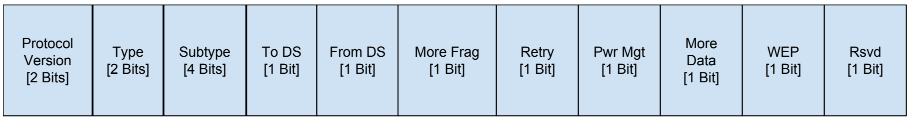
The packet format specified for 802.11 is depicted in the following figure:



Frame Control [2 Bytes]	Duration ID [2 Bytes]	Address 1 (source) [6 Bytes]	Address 2 (destination) [6 Bytes]	Address 3 (rx node) [6 Bytes]	Sequence Control [2 Bytes]	Address 4 (tx node) [6 Bytes]	Data [0 -2312 Bytes]	FCS [4 Bytes]

FIGURE 3.11: 802.11 MAC packet Format.

The MAC packet begins with 2 Bytes of Frame control. This field is subdivided in several subfields used to determine the following parameters:



Protocol Version [2 Bits]	Type [2 Bits]	Subtype [4 Bits]	To DS [1 Bit]	From DS [1 Bit]	More Frag [1 Bit]	Retry [1 Bit]	Pwr Mgt [1 Bit]	More Data [1 Bit]	WEP [1 Bit]	Rsvd [1 Bit]

FIGURE 3.12: 802.11 MAC Frame Control field.

Most of the names are self-explanatory. The protocol version indicates which version of standard is being used that is 0 for 802.11. The Type and Subtype fields are joined together to create a 6 bits frame type/MAC service identifier. The two next fields, from to DS from DS, indicate if the packet comes from or goes to the Distribution System (link between APs).

More fragments indicate if the frame is followed by more fragments and retry is set when the frame has been send more than once. The power management field is set to indicate that the station is going to power save mode and More Data is used by the AP to indicate to a station that has more data stored to be delivered. WEP indicates that the frame body has to be processed by the WEP algorithm. The last bit is reserved for future use.

Then it comes the duration ID that indicates for how long will the transmitter wait the reception of the ACK. After that, the addresses field come. The first address indicates the 48-bit receiver MAC address. The second one is the transmitter MAC address. The third address is used for filtering by the receiver. The fourth is only used when a DS is involved and the network is used to interconnect bridges.

The sequence control 2-byte field is used to filter and discard duplicate messages. After that comes the message and finally a 4-byte checksum.

3.4 WiFi incompatibilities for long range operation

The 802.11 standard is intended to offer wireless LAN access, that is, in a limited area as home, office, laboratory, etc. Thus, the protocol and its internal mechanisms as the MAC layer are optimized to be used in this situations. Particularly the MAC layer has some constants that are incompatible with long range operations, specially the ACK timeout.

As explained, the 802.11 MAC layer implements a positive ACK for each send packet. If no ACK is received after a certain amount of time, the packet is considered lost and the 802.11 MAC layer will initiate a retransmission. For that reason, the ACK waiting time imposes a maximum range of operation between the transmitter and receiver.

Although the ACK timeout value is not a part of the standard, the IEEE recommends to set this value to 1μs for 802.11b devices. Taking into account that electromagnetic waves travel at $300 \times 10^8 \mathrm{~m/s}$ this sets a maximum operative distance between devices of 300 m.

Even though the value of the ACK timeout could be increased, there are usually some upper limits set by the manufacturers [46], which are much lower than the ones needed for a LEO satellite link.

A sufficient large ACK timeout would indeed solve the problem but it would have a really negative impact in the system's throughput because the transmitter would have to wait the long ACK timeout to realize from packet losses.

3.5 Long range operation WiFi studies and projects

The unlicensed use of the spectrum jointly with the possibility to obtain high throughput with cheap devices has motivated the research to apply WiFi for long range point-to-point communications with different purposes. The first works for adapting WiFi to long link distances focused on facing the problems on the default protocol and proposing solutions [47]. Paralely, extensive performance characterization of long range WiFi links was done [48], [49]. After that, several projects which deployed long range networks were done:

RoofNet [50] developed in 2002 at the MIT was a multi-hop wireless access network. It was based on 802.11 nodes that were self-managed. The principal modifications on the WiFi standard included the use of proprietary operation mode provided by Prism chips and the use of PCs to perform routing. Nodes separated by long distances could communicate by routing the packets through the mesh network. As the node-to-node hops were of short distance, this project didn't suffer from ACK or RTS/CTS timings. However, the achieved distances were moderate and the bit rate low.

WiFi-based Long Distance Networks (WiLDnet) [51] was designed at the Berkeley University with the objective of replacing traditional network connectivity products as cellular or satellite communications with WiFi. The developing world can highly benefit from using long range WiFi networks because 802.11 devices are cheap and their use straightforward. WiLDnet modified the layer 2 of the 802.11 protocol in order to have bulk acknowledgments with a Hybrid-ARQ implementation. Moreover, it modified the CSMA/CA multiple access method by TDMA. Both modifications are necessary to obtain high throughput in links were the packet error rate is elevated and the propagation time non-negligible. They used firmware modifications to obtain the desired changes.

Since 2015, the open-source Wifibroadcast project [52] is available on the Internet providing live transmission of HD video over WiFi adapters. It uses packet injection in transmission and monitor mode in reception. In addition, it supports nice features as diversity and Reed-Solomon erasure coding. However, it is focused on provide unreliable low latency video transmission for multicopters. For that reason, it would fail to provide a reliable downlink.

On the 7th of July 2015, the Soarex-8 Flight [53] from the NASA Ames Research Center successfully tested an experimental payload. It consisted in transmitting video from space using WiFi devices with packet injection and monitor mode. They set a new distance record for WiFi and shown that long range communications using WiFi products are feasible. The system used a repetition code by sending each packet twice, the simplest form of error correction, which was good enough to stream video but that would fail to reliably transmit data files.

3.6 Custom Solution

As it has been explained the main problem of using WiFi for long range links is the ACK timeout. The solution passes for bypassing its use. The following subsections describe the possibilities that were studied and the selected solution.

3.6.1 Broadcast/Multicast

The 802.11 standard indicates that broadcast or multicast packets are not acknowledged to avoid broadcast storms. If a broadcast packet had to be acknowledged by all the receivers the network would had to deal with an enormous amount of data created by massive ACK responses.

One solution could come by means of using only broadcast or multicast packets, although in order to send broadcast or multicast packets the transmitter has to be associated to some network. That implies a series of control messages that are not desirable to have.

3.6.2 802.11e

Similarly to the previous solution, another tweak that can be done in order to avoid the ACK is to use the 802.11e standard. This standard was issued to give support to bandwidth sensitive applications as voice and video. Without this implementation the 802.11 networks are highly inefficient treating this kind of data [54].

The 802.11e standard defines policies to be applied, one of these rules is the NO-ACK policy. In principle this directive should be applied to time-sensitive data, but it could be used to transmit any kind of packet. Again, as the previous solution, the transmitter and receiver should join a network before transmitting and that implies sending control messages.

3.6.3 Injection + Monitor

A non-ordinary solution is proposed in this section. Packet injection is used by researchers and security experts to test the 802.11 protocols. It is commonly used for penetration testing because custom raw data frames can be send without the need of being connected to the network.

RFMON or monitor mode is a mode in which the wireless adapters monitor all traffic received from the wireless network independently of the destination address.

The combination of these two modes has a lot of potential to bypass all the obstacles that the vanilla WiFi protocol introduces for long range links, and allows the following key points:

- No ACK or other associated timeout problems.
- Unidirectional communication without standard control packets (one-to-many communications).
- Forge custom packets.
- Ubiquitous support of injection mode under Linux systems.

The downside is that not all the wireless adapters are compatible with these modes or have limited support of data rates under injection. For that reason, the selection of the appropriate hardware will play an important role on the construction of the communications system.

3.7 COTS Hardware Selection

3.7.1 Selection Process

A thorough selection process was carried over an extensive list of WiFi devices available on the market in order to identify the most suitable one to be used. The extensive list of candidates was filtered according to the following characteristics:

USB Connection

WiFi adapters may work with data flows up to hundreds or thousands of Mbps. For instance, the 802.11n standard has a theoretical maximum throughput of 300 Mbps and the 802.11ac standard increases this value to 3 Gbps. Wireless adapters come with different buses to support this data rates. For instance, WiFi adapters designed for desktop computers usually come with a PCI/PCIe bus and for laptop devices they may come integrated or feature a PCMCIA bus. Another flavour of WiFi adapters come with USB connection, what makes them portable and plug-and-play.

An USB WiFi adapter will be selected as it is the only bus compatible with the characteristics of the nanosatellite's onboard computer. Due to the expected received power from the satellite link only robust modulations will be used, meaning that the maximum bit rate is going to be of several Mbps. Consequently, a USB 2.0 bus will be enough to fulfill the connectivity requirements.

Standards

The family of 802.11 standards use different modulations, coding rates or even bandwidths. However, standardization bodies have made an effort to guarantee backward compatibility when issuing new standards. Adapters that incorporate new standards are usually able to adapt to any former standard release. For instance, 802.11n transceivers tend to be compatible with the 802.11g/b standards, as it has been seen in 3.2.4.

One of the key elements to obtain the most of the satellite link is to use adaptable bit rate. The possibility to select between a large variety of modulations is a big plus, as it will permit a finer grain tuning. In order to have all the modulations available for the 2.4 GHz band (ignoring the 802.11ac ones), the transceiver has to implement the 802.11n standard and be backward compatible with 802.11b/g standards.

Reception Sensitivity

The reception sensitivity is an important parameter to take into account as it will have a direct impact on system link budged. The sensitivity of an adapter not only depends on the WiFi chipset itself, but also on the RF components of the PCB.

The selected adapter should be able to work with the weakest signal strength possible, paying special attention to the performance at low bit rate modulations.

Output Power

Although the downlink system is expected to have a powerful amplifier it is desirable that the adapter itself has enough output power to be used as a pre-amp.

There are in the market a lot of WiFi adapters labeled as "high-power" of dubious credibility, so a careful and contrasted inspection should be done.

Size

Nanosats are very constrained in dimensions and every cm^3 has to be taken into account. When seeking for the most suitable WiFi adapter the physical dimensions may be a decisive point. For instance, adapters larger than 8 cm may have problems to fit into a 1U cubesat.

Injection/Monitor support and open Driver/Hardware

A WiFi card may inject packets if it has drivers and firmware that are compatible with traffic injection. Linux-based OS are widely used when performing network analysis due to the great availability of open-source wlan drivers. In fact, most of the Linux distributions come with out-of-the-box support of injection and monitor mode for a large list of WiFi adapters. Although the exact number of WiFi devices that support injection under Linux is hard to determine with exactitude, the list raises above the one-thousand adapters [55].

Last but not least, the availability of open firmwares is a major plus to take into account in the adapter's selection. An open firmware allows the programmer to understand how the hardware works, to deeply go into the details and opens the possibility to make modifications to the default mode of operation.

Although most of the drivers in Linux are open-source, the availability of open firmwares is scarce, which heavily-decreases the candidates list.

3.7.2 Candidates

From the myriad of devices available on the market only three candidates comply with all the filters: ALFA Network AWUS036NHA, TP-LINK TL-WN722N and Ubiquiti Networks WiFiStation EXT. The list could be increased to 10 or 15 more candidates if the requirement of high output power was dismissed.

TABLE 3.6: Comparison between WiFi adapter candidates, data obtained from [56],[57] and [58]

Adapter	Standards	Sensitivity	Output	Size
AWUS036NHA	b/g/n	-96dBm @ 1Mbps -91dBm @ 6Mbps -91dBm@ 6.5Mbps	27dBm	6.3x8.5 cm
TL-WN722N	b/g/n	-90dBm @ 1Mbps -88dBm @ 6Mbps -86dBm @ 11Mbps	20dBm	9.4 x 3cm
WiFiStation EXT	b/g/n	-97dBm @ 1Mbps	30dBm	13.6x4 cm

At first glance, the WiFi adapter that better fits with the desirable characteristics is the Ubiquiti Networks WiFistation EXT due to its higher sensitivity and greater output power. However, the manufacturer has abandoned its production and nowadays this product is deprecated. In addition, the size of this product is too large to fit inside a 1U cubesat.

Discarding the Wifistation there are two options left, the AWU036NHA and the TL-WN722N. The selected adapter will be the AWUS036NHA from Alfa Networks as it claims to have better sensitivity and has a form factor suitable for the nanosat integration.

3.8 Alfa Network AWUS036NHA

Packaged in a fairly big box and labeled as a "long-range" WiFi product surrounded by flames, there is a clear marketing intention to attract costumers attention. The default version comes with a 5 dBi omnidirectional antenna as well as with 1.5 m of USB cable and some stickers. With a price of around 20 euros it can be found on most of specialized consumer electronics shops.

3.8.1 Technical Specifications

Alfa Network AWUS036NHA is a USB WiFi adapter 802.11b/g/n certified, capable of offering a maximum throughput of 150 Mbps. It features an Atheros AR9271L chipset [59] for baseband processing and a Skyworks 2576L power amplifier [60].

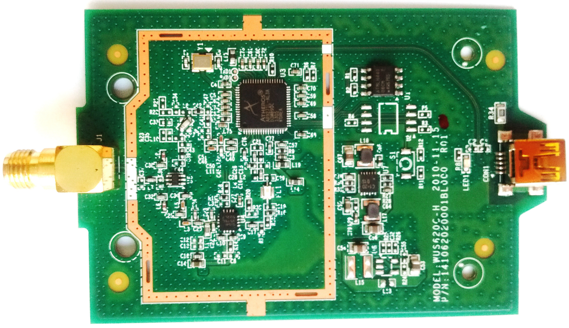


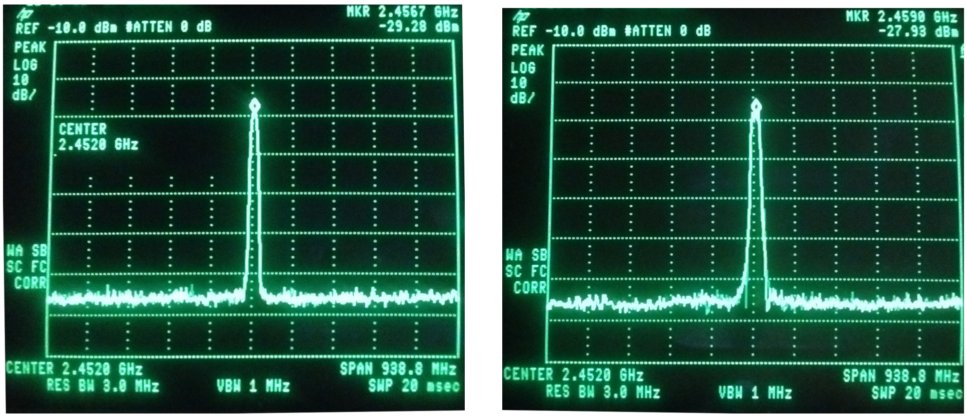
FIGURE 3.13: ALFA AWUS036NHA PCB.

The picture shows that the PCB has been prepared to accommodate a RF case which hasn't been included, probably for cutting down manufacturing costs.

Output Power

The ALFA's datasheet says that it is capable of providing 27 dBm of output power while working with 802.11g/n standard and 29 dBm in case of the 802.11b. In principle, these values are feasible as the power amplifier integrated in the PCB says that it features 26 dBm of output power when operating on the 802.11g/n standard and up to 29 dBm while working on the 802.11b.

In order to test the real output power of the device, the RF output of the WiFi adapter was connected to the input of a spectrum analyzer using a 50 dB attenuator in between:



(A) Output power for DSSS modulations.

(B) Output power for OFDM modulations.

FIGURE 3.14: Output power for DSSS and OFDM modulations.

As the figure shows, the measured output power for both cases is very similar. Taking into account the 50 dB of attenuation, that gives an output power of 21 dBm for DSSS modulations and 22 dBm for OFDM.

3.8.2 Open Firmware and Driver

Linux Wireless Networking Architecture

The following picture describes how Linux manages the WiFi devices.

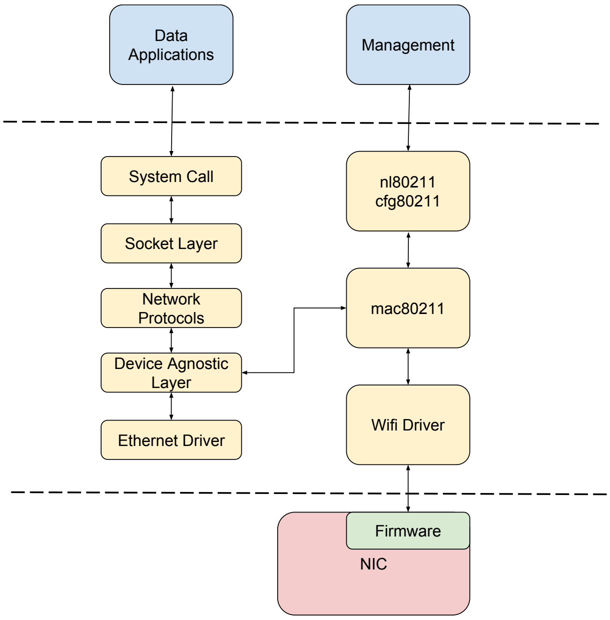


FIGURE 3.15: Linux wireless networking architecture, inspired from [61].

The figure is divided in three sectors: The upper part painted in blue is the User Space, the memory zone where the processes run and code not related with kernel is located. At the middle of the figure the Kernel is represented in yellow, this portion of memory is restricted to the operating system and the services that it offers. Finally, the bottom part in red represents the external hardware, where the NIC pertains to.

The 802.11 standard defines the physical and MAC layer. On the one hand, the physical layer defines how the signal processing has to be done. As it has to perform real-time operations it is always implemented in hardware. On the other hand, the MAC layer may be also fully implemented in hardware, but it is also possible for non time-critical functions (most of the MAC layer) to be implemented in software.

Devices in which the MAC layer is implemented in hardware are called full MAC whereas devices that have most of the MAC implemented in software are referred as soft MAC devices. Nowadays most of the WiFi adapters are soft MAC, they implement the physical layer in silicon and a large part of the MAC layer in software.

Following a top-down analysis of the figure, let's imagine that an application located at the user space is willing to send messages to the WNIC. As the picture shows, there are two paths in which the applications of the user space can communicate with the wireless adapter. The path of the left is used for data applications (web browser, torrent client, email client, etc) whereas the right path is reserved for management applications (wi, wpa supplicant, network connection manager, etc).

When a data application wants to send information to a NIC it invokes a socket indicating the port and the protocol used. This socket calls a system call interference and from that moment on, the connection is managed in the kernel space. Then, the information is passed to the socket layer, here is where actually the socket is created using the parameters from the application that has requested the socket. Next, the socket layer passes the information to the networks protocol box, that will be in charge of forging the packets and maintaining the end-to-end communication according to the protocol defined by the application's socket. When the packets are ready to be send, the device agnostic layer is the one in charge of delivering the packet to the device driver.

At this point there would be two options, if the socket has been opened on an Ethernet interface the device agnostic layer would dispatch the packet to the Ethernet driver. Alternatively, if the data is intended to be send through the WiFi adapter, the device agnostic layer will convey the data to mac80211 that will pass the packet to the WiFi driver. The driver is responsible to send the information to the NIC in a way that the adapter's firmware understands it.

Out of the Kernel space and inside the WiFi adapter the firmware is located, it is a piece of software that interacts directly with the hardware and with the kernel's driver.

As mentioned before, the right data path is reserved for management applications. In that case the applications located in the user space communicate with the kernel by means of a netlink socket controlled by nl80211 that resides inside the cfg80211 block. That block is able to directly interact with the mac80211.

Management applications are in charge of non time-critical functions as re-association, scan, authentication, etc. Control frames that do are time-critical are managed by hardware.

Atheros Open Firmware and Driver

As it has been seen previously the two elements that are closer to the NIC are the firmware itself that runs on top of the hardware and the driver that is the link between the device and the kernel.

Most of the manufacturers only release binaries of their firmware. For that reason, even if the driver is open source the modifications that can be done on the 802.11 MAC layer are limited.

The reason why manufacturers are reluctant to offer the code of their firmware is because they could infringe the law. With an open firmware the default configuration of the 802.11 compliant devices could be hacked and could potentially cause problems to other devices.

Atheros was the first company that decided to open the capabilities of their hardware to the users. The WiFi cards that the company manufactured were able to work under a wide range of frequencies and lots of radio parameters were able to be controlled by software. As a tradeoff between offering the full potential of their cards and avoid law problems with the standaritzation and RF agencies, they wrapped the hardware registers with a Hardware Abstraction Layer. This HAL was protecting the critical register of the hardware while letting the users play with other ones. By that time a popular family of drivers for network enthusiasts was born from the MadWiFi project [62], these drivers that the HAL provided by Atheros.

In 2008 Atheros decided to open-source the code of their HAL, making their hardware the first one to be fully supported by open-source software. Atheros continued being pro-active with the open-source community and released a fully open-source driver "ath9k" for their 802.11n cards the same year.

In 2013 Atheros made available on GitHub an open-source firmware for two of their chipsets: AR7010 and AR9271 [63]. On the one hand, AR9271 is a SoC solution that incorporates all the WiFi functionality inside the chip with 802.11n and USB support. On the other hand, AR7010 is usually paired with AR9287 chip to provide full functionality. While the AR9287 is in charge of the baseband, MAC and MIMO processing the AR7010 is used offload the host-CPU usage by computing 802.11n cpu-instensive functions [64].

With this firmware release, dozens of wireless adapters were suitable to be modified and opened the possibility to researchers, enthusiasts and hobbyists to learn and play with 802.11 hardware.

3.9 802.11 Modifications

3.9.1 Injection and Monitor Mode

As explained, in order to avoid control packets and have a truly unidirectional communication injection and monitor mode will be used.

The packet injection and packet capture in monitor mode will be performed with the support of the *libpcap* [65] portable C/C++ library for network traffic capture. The following figure shows the interaction of *libpcap* with the other elements of the OS:

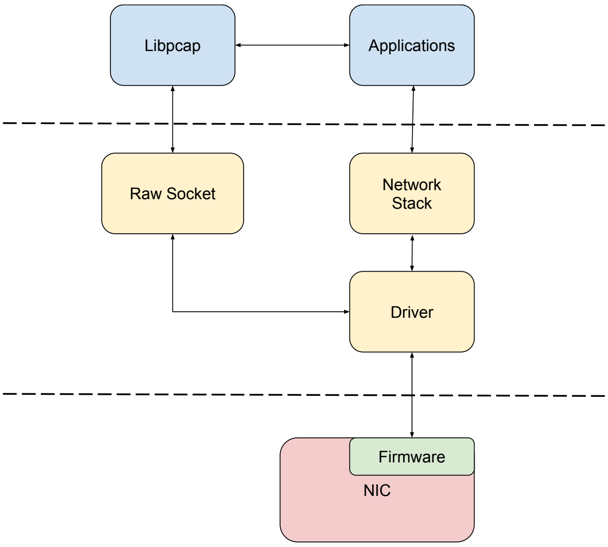


FIGURE 3.16: Libpcap and Linux networking structure, inspired from [66].

Libpcap is an open source library that provides a high level interface to network packet capture systems [65]. The main goal of its developers is to offer a plataform-indepedent API for capture applications.

When the Kernel receives packets captured by the adapters it passes the information through the networking stack, that is in charge of stripping the packets, inspecting their headers and apply the protocols procedures. After that, only the payload of the packets is sent to the applications allocated at the user space, making the underlying layers invisible.

In order to access to the lowest possible level, the *libpcap* library grabs directly from the driver the packets, bypassing the networks stack and capturing the entire packet with headers included. To do so, the library creates a raw socket that communicates with the adapter's driver. When the driver receives a packet, the library forwards it to the user space where applications can use it.

To capture all the packets emitted on a certain frequency without being connected to any AP it is necessary to put the device in monitor mode. This mode has to be supported by the driver and is activated by managing tools as $iwconfig$ or iw that use the management application data path.

The injected packets follow the same path but in inverse direction than the received packets in monitor mode. The packets are forged in the application space including the packet's headers. Then, *libpcap* opens a raw socket that passes the packets to the driver. Finally the driver forwards the entire forged packets to the network adapter to be transmitted.

Using packet forging and injection along with monitor mode the user gains full potential on every bit that has been transmitted and received.

RadioTap Header

Radiotap is a "de facto" standard for 802.11 frame injection and reception. The Radiotap header format is a mechanism to supply additional information about frames, from the driver to user space applications such as *libpcap*, and from a user space application to the driver for transmission [67].

For example, when the driver receives a new packet from the adapter, it also obtains information about the power it was received, in which channel has been transmitted, which modulation has been used, etc. All this information is usually never forward towards upper layers. The RadioTap header is a way to send this information to the user space where it can be used by applications. It can also be used the other way around, to send jointly with a forged packed additional information e.g. which antenna has to be used for transmission.

3.9.2 Firmware

The firmware for the AR7010 and AR9271 USB 802.11n NICs is available in an open-source fashion at Github [63].

The code can be downloaded, modified, compiled and sent to the adapter every time it is plugged-in. In principle, most of the 802.11 MAC can be modified, although there may be limits set by the hardware. The modifications are usually not straightforward and a good knowledge of the underlying hardware structure is needed.

When an adapter supports packet injection usually uses the lowest and more robust possible modulation which is 1 Mbps. In order to customize the injection rate a simple modification has been made to the firmware [68] set the bit rate a the fixed desired data rate. Several firmwares have been compiled, each one working on a different bit rate. When the transmission rate wants to be changed, the appropriate firmware has to be loaded into the adapter.

3.10 Conclusion

Chapter 3 gives a detailed insight into the layer 2 and layer 3 of 802.11-family standards. The physical modulation may be divided between DSSS and OFDM achieving bit rates that range from 1 Mbps to 300 Mbps.

Packet injection and monitor mode has been chosen to bypass the ACK timeout that limits the operational range of WiFi devices. The ALFA WiFi adapter has been selected from an extensive list for its high output power, great sensitivity, small form factor and the availability of open source drivers and firmware. Finally, the packet forge, monitor mode and firmware modifications have been explained.

Chapter 4

Combating Fadings and Loss-of-Signal

As it has been seen in chapter 2, the WiFi downlink has almost no margin over the sensitivity level of the transceivers. Moreover, the scintillation effect and the pointing errors will introduce losses that are expected to create outage periods of several seconds.

Having no access to the physical layer for being it implemented in hardware, the link robustness will be achieved by applying LDPC-Staircase and Reed-Solomon codes at application layer.

This chapter explains the history of error correcting codes paying special attention to the Reed-Solomon codes. Then, the concept of loss-resilient codes is introduced explaining the unfeasible computational application of Reed-Solomon codes for large data. After that, the invention of fountain codes is explained with the examples of the Luby Transform and Raptor/Q codes. The computational cost of these implementations is contrasted with the LDPC-Staircase Codes which turn out to be well-suited for the characteristics of the nanosat satellite link.

4.1 Reed-Solomon Codes

4.1.1 History

In 1948 Shannon stated that error-free transmission is possible as long as the transmitter does not exceed the channel's capacity [69]. Fueled by this promising result, information theory scientists began seeking for error correcting codes with optimum performance.

In 1949 Golay [70] published a linear correcting code capable of correcting any three errors out of 23 bits, being 12 bits of information and 11 bits of parity. Hamming codes [71] were discovered in 1950 and provided the capability of detecting and correcting errors in reception. A popular Hamming code was the "(7,4)" which for each 4 bits of information it adds 3 bits of parity being able either to correct a single-bit error or detect up to two errors.

The Reed-Solomon block codes were published in 1960 [72] by Irving S. Reed and Gustave Solomon. Differing from Golay and Hamming codes, they encode groups of bits using Galois fields arithmetic. The original paper described the guidelines for encoding and decoding information. Although the encoding was straightforward, the decoding mechanism that the original paper proposed was impractical. The Berlekamp [73] algorithm introduced in 1968 and the modifications made by Massey [74] drastically reduced the computational cost of the Reed-Solomon decoding, making them suitable for real-world applications.

The block-encoding fashion makes Reed-Solomon codes robust against burst errors that typically happen in telecommunications channels. Moreover, they pertain to the maximum distance separable codes family, attaining optimal error detection and correction capabilities for a fixed message and parity length [75].

Reed-Solomon codes have been applied to many applications. In 1977 two spacecraft were launched to follow a cosmic journey, namely Voyager 1 and Voyager 2. The two probes took advantage of an alignment of the outer planets for visiting them. Reed-Solomon Codes were used to protect the information and for effectively decrease the BER of the communications system which was working at very low SNR. Reed-Solomon codes have been also used to daily-day consumer technologies as CD, DVD or Blue-Ray. The DVB also uses this Reed-Solomon as well as xDSL or WiMax [76].

4.1.2 Characteristics

Reed-Solomon is a block encoder that works with groups of bits. These groups are called symbols and usually, due to microprocessor's architectures, the symbol length is chosen to be of 1 Byte (8 bits). A message encoded with Reed-Solomon Code of length n and k symbols of information has $n-k$ symbols of parity. The number of recovery symbols is user-adaptable and determines the error detection and correction capabilities of the Reed-Solomon code:

$$ErrorDetection = n - k, \tag{39}$$

$$ErrorCorrection = \lfloor n - k/2 \rfloor . \tag{40}$$

Usually, an even number of recovery symbols is used because as the floor operand indicates, error correction capabilities only increase in even numbers. The following figure shows the structure of the Reed-Solomon Code when is applied to a message of length $n-k$:

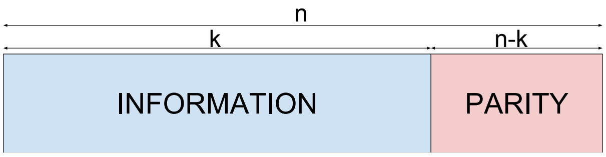


FIGURE 4.1: Reed-Solomon Code structure.

The more parity symbols are added the more errors the Reed-Solomon code may detect and correct. However, the decoding complexity quickly increases with the number of parity symbols. For deep space communications there is the Reed-Solomon standard (255,223) proposed by the CCSDS in which there are 223 symbols of information and 32 symbols of parity in a 255 encoded packet [77]. The packet length of this standard is 255 symbols because it works over Galois fields of dimension 2^8 and the 32 symbols of parity permit the detection and correction of any 16 erroneous symbols in the packet.

Although Reed-Solomon codes have been extensively used in satellite communications, presenting good burst error correction properties, they can't combat loss of signal. To achieve robustness against loss-of-lock without having feedback and consequently without ARQ mechanism the so-called rateless codes are going to be explored.

4.2 Loss-Resilient Codes

4.2.1 Reed-Solomon

The basics of Reed-Solomon codes may be used to create ideal Maximum Distance Separable (MDS) codes [78], [79]. When the Reed-Solomon knows where the erasures have happened it is able to recover:

$$ErasureRecovery = n - k, \tag{41}$$

being now n the total number of packets and k the number of information packets. This way, for each received repair packet the Reed-Solomon code can recover one of information. In other words, when Reed-Solomon is used as an erasure code it is able to obtain the original message when the number of received packets, independently if they are information packets or repair packets, is equal or greater that the number of packets that conformed the original information. The next figure depicts how the Reed-Solomon code works when used in erasure mode:

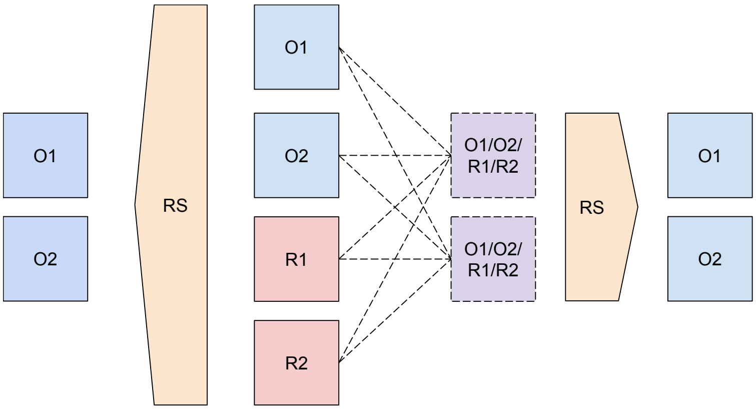


FIGURE 4.2: Reed-Solomon Code erasure mode.

The diagram shows how two packets of information $O1$ and $O2$ are encoded with Reed-Solomon with a coding rate of 0.5 and two more redundant packets $R1$ and $R2$ are forged. After that, the packets are send through the wireless channel that introduce packet losses. At reception, only two of the 4 Reed-Solomon packets (01, 02, $R1$ and $R2$) are received. However, as the coding rate has been 0.5, independently of which two packets have been received the decoder is able to obtain the original data $O1$ and $O2$.

The main problem of the Reed-Solomon codes for erasure recovery is their encoding and decoding complexity when the number of packets to be encoded is large. In fact, when the size or number of packets in which the information is split grows the computational cost becomes brutally expensive and far away from any real-world implementation. Consequently, Reed-Solomon codes can only be used when the number of repair packets and the packet sizes are low.

As the images to be downloaded will be around 100 MB and the packet size no less than several hundreds of bytes, the number of source packets to be codified would be of around 100000. This figure is absolutely not achievable for Reed-Solomon codecs in nowadays computers.

4.2.2 Fountain Codes

Fountain Codes or Rateless Codes, are codes with the capacity of generating a limitless number of encoding symbols from a given set of source symbols. Ideally, the original source symbols can be recovered if the size of received symbols is equal to the size of source symbols.

However, practical implementations need to receive more symbols than the number source symbols to achieve a complete recovery. Given an input of k symbols, fountain codes are able to decode them with $(1+\epsilon)\cdot k$ received symbols, where ϵ is the recovery overhead and $\epsilon > 0$ for real codes and $\epsilon = 0$ for the ideal case.

Fountain codes work at packet level and imply a change in the classical communications paradigm, only the amount of received information matters, and not in which order or what parts have been received.

The following images exemplify the changes on the communications scheme when using Fountain codes. On the one hand, we can observe that classical approaches using ARQ ask for retransmission of corrupted packets:

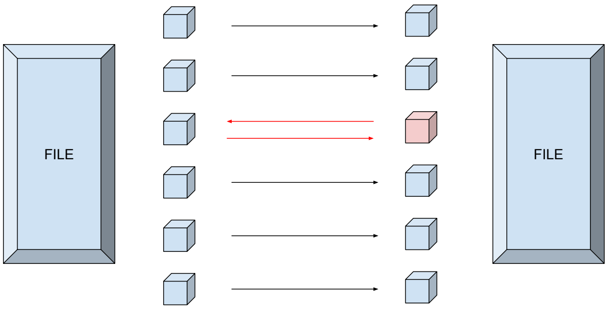


FIGURE 4.3: Automatic Repeat Request (ARQ) error control mechanism. The red packet was received with errors and the receiver, in order to obtain the file, has to request a retransmission.

On the other hand, the use of Fountain codes makes unnecessary the feedback channel, as only the volume of received packets matter:

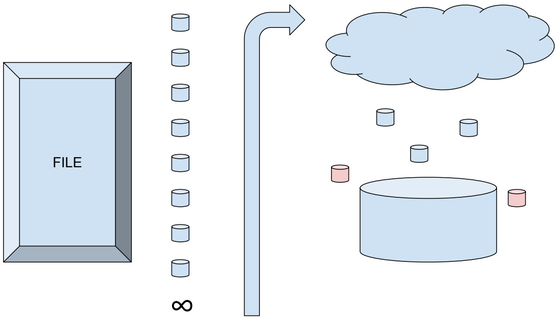


FIGURE 4.4: Fountain codes mechanism. The receiver receives packets or not, at the end, only how many packets are received is relevant.

Ideal Fountain codes seem to be tailored for obtaining a robust unidirectional link in which losses of packets are expected. As only the amount of received information is relevant and the transmitter may generate infinite packets for any source file, the onboard computer could encode the multispectral images using fountain codes. The ground station would receive packets and stop when enough information would had been recovered.

Fountain codes were theoretically proposed by Byers [80] in 1998 and during the following years practical implementations were developed. Although some quasi fountain codes have almost reached optimum performance in terms of recovery overhead ($\epsilon \approx 0$), some of them exhibit high computational cost than wouldn't be affordable for the onboard computer.

Luby Transform

The first practical implementation of a fountain code was proposed by Luby in his paper published in 2002 [81].

Luby transform is a rateless code, meaning that it doesn't use a predefined coding rate and the symbols can be generated on the fly at the transmitter side.

The Luby transform code is based on a relatively easy algorithm that involves XORs for encoding and decoding. The basic principle is to spread the information stored in each one of the k packets of the source file.

The encoding process is based on XORing blocks and appending identifiers:

1. The source file is split in k blocks.
2. A random number d is generated, $1 \leq d \leq k$ indicating how many k source blocks will be XORed, also known as the degree of the packet.
3. A number d of blocks are randomly chosen and a XOR operation applied on them.
4. The identification of the blocks that have been XORed and the k total number of blocks is appended to the obtained XORed block.

The decoding is done by combining the received blocks and XORing them appropriately to obtain the k source blocks. The main objective is to reduce the degree of the blocks until it reaches 1:

1. When a packet is received the block identifiers that contains are retrieved.
   (a) If the degree of the block is 1, then as the block hasn't been XORed the original information is obtained, and it is saved in a pool of decoded blocks.
   (b) If the degree of the block is greater than 1 the block is processed with the decoded blocks to lower its degree. It the resulting block still has a degree greater than 1, it is moved to the pool of undecoded blocks. If the degree has been reduced to 1, it is placed in the pool of decoded blocks.

   This process is repeated for every new received packet, decreasing the degree of the undecoded blocks until all of them reach 1, and the message is obtained.

The following figure exemplifies a naive example on how the Luby transform work:

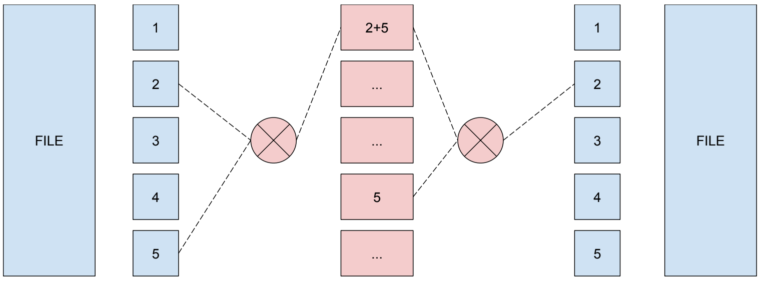


FIGURE 4.5: Luby Transform scheme.

The figure shows a simple example of an interation of the Luby transform. The file has been split in $k = 5$ parts and the random number generator has obtained the value $d = 2$. Three random blocks have been selected: 2 and 5 and XORed to obtain the $2 \oplus 5$ block.

At reception the receiver had the packet 5 already in the pool of the decoded blocks and so, it does $(2 \oplus 5) \oplus (5) = 2$ to obtain the block 2.

The complexity of the algorithm is placed on the random distribution the pseudo-random number generator has to follow for selecting the appropriate d value. Luby proposed the robust soliton distribution which presents great properties to maximize the recovery probabilities with minimum overhead.

The main problem of the Luby transform is its high complexity in encoding and decoding having a computational cost of $O(k \cdot log(k))$ which is not acceptable for high values of k. In order to solve this problem, the Raptor and more recently RaptorQ codes were invented.

Raptor and RaptorQ Codes

Raptor codes [82] were invented in 2000 by Shokrollahi as a solution to decrease the encoding and decoding complexity of Luby transform codes.

Raptor codes characteristics are close to ideal fountain codes and are nowadays part of many standards as 3GPP MBMS, DVB-H or IPDC. Raptor Codes are being currently replaced by RaptorQ codes, an improved version that enhances the recovery capabilities for low ϵ.

The main idea behind Raptor codes is to apply pre-coding to relax the condition that all input symbols need to be recovered. If a Luby transform code needs to recover only a constant fraction of its input symbols this allows linear time encoding. All the source symbols can still be recovered from a part of the intermediate symbols if during the pre-coding a traditional erasure correcting code has been applied [83].

The following image exemplifies the encoding process:

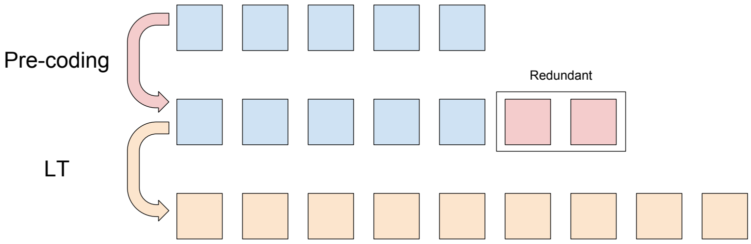


FIGURE 4.6: Raptor Code encoding process.

Raptor codes achieve linear complexity when generating source or repair symbols from intermediate symbols. For this reason, most literature says that Raptor codes achieve constant encoding/decoding time. However, it is worth noting that the generation of the intermediate symbols has cubic complexity and it is related to the number of symbols to forge [84]. That is usually not a problem because the number of symbols to be generated is low. For example, generating several hundred symbols may be achieved with few milliseconds in low-end microprocessors. However, for applications that need to encode a large quantity of symbols Raptor Codes become impractical. As commented before, around 100000 symbols would be needed for transmitting $\approx$ 100 MB images in 1500 B 802.11 packets, this number of symbols makes the use of Raptor codes computationally impractical for the purposes of the link.

Moreover, although there are some implementations of the Raptor codes available on the web [85], [86] is worth noting that some techniques that Raptor codes use are patented by Qualcomm [87].

Although the properties of a fountain code are definitely desirable for a robust unidirectional link they are not a must when designing a LEO link. As the satellite pass-by will only last for a few minutes, the benefits of generating an infinite number of repair blocks are not used because the number of packets that may be send is limited. In fact, a loss resilient code with the capability to generate enough redundancy is sufficient.

4.2.3 LDPC-Staircase Codes

LDPC codes were discovered by Gallager in 1960 [88], but were forgotten for most than 30 years until they were rediscovered. LPDC codes are based on spare bipartite graphs and their encoding process is really fast. They have recently received special attention because they are able to approach capacity with fast decoding iterative methods. They have been used in broadcast standards as DVB-S2, in home networking as in G.nh or in 10 GBase-T Ethernet standard.

LDPC-Staircase codes are a subfamily of LDPC codes in which the encoding is trivial as it consists of matrix multiplications. This gives to the LPDC-Staircase codes linear encoding time. The decoding using Maximum Likelyhood achieves maximum performance at the expense of having cubic complexity. Iterative decoding may be used in order to obtain linear decoding time, but sometimes it fails resolving the system when decoding with low overhead.

LDPC-Staircase is an erasure code that meets all the requirements for a nanosatellite unidirectional link. The encoding process is trivial and fast enough to encode high-speed links with modest CPU's. The complexity is moved towards the decoder, that is placed at the ground station where computing resources are not limited. LDPC-Staircase is able to generate enough redundancy to ensure reliability without compromising the recovery performance, being it able to achieve figures that are close to Raptor codes [89].

Last but not least, there is a collection of open-source, IPS-free (with no patent infringements) codecs called OpenFEC [90] that contains a marvelous implementation of a LDPC-Staircase codec able to compete with commercial Raptor codes solutions. The implementation of this codec is standardized in the RFC 6816 [91].

4.3 System Coding

LDPC-Staircase codes alone are able to protect against loss-of-signal that produce packet losses. However, LDPC-Staircase codes are erasure codes and not error correction codes, meaning that if the packets are received with errors they are treated as lost. In order to make the system also robust against bursts of errors a Reed-Solomon error correction code may be used as outer code for the LDPC-Staircase code.

However, this can only be done if in reception the Kernel is able to pass to the user-space packets that contain errors. When a 802.11 packet is forged the Kernel introduces a 4-Byte FCS, as it has been seen in figure 3.11. In reception, the networking stack checks the FCS of the incoming packets and automatically discards the ones with bad FCS. However, in Linux it is possible to use control applications as iw to set the reception flag "fcsfail", this way frames with FCS errors will also be delivered to the user space.

There are plenty of different Reed-Solomon implementations available on the Internet. The RSCode project [92] Reed-Solomon implementation has been selected because it is open-source, it has GPL-licensed and it features a user-selectable parity length and Berlekamp-Massey decoding algorithm. As both OpenFEC LDPC-Staircase codec and RSCode Reed-Solomon implementation are written in C the integration of them into the same project is straightforward.

4.3.1 LDPC-Staircase structure

The LDPC-Staircase code needs to know the following parameters in reception:

In general:

- Symbol size.
- Total number of generated symbols, n.
- Number of source symbols, k.

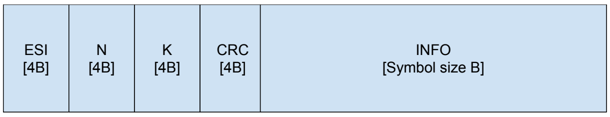
For each packet:

- Encoding symbol identifier, ESI.
- Integrity check, CRC.

The symbol size will be static and can be hardcoded in the reception program. The n and k parameters only need to be received once during the reception of the packets. However, in order to be sure that this crucial parameters are received, they are included in every packet as well as the ESI.

When a packet is received with errors (FCS fail) the Kernel passes all the data to the user space. Then, the program may apply the Reed-Solomon correction to the packet that contains errors to repair them. In order to know if the recovery has been successful or not a CRC has to be appended to every packet. The CRC code used has public domain license and is simple and fast [93].

The structure of LDPC-Staircase is as follows:



ESI [4B]	N [4B]	K [4B]	CRC [4B]	INFO [Symbol size B]

FIGURE 4.7: LDPC-Staircase Frame.

4.3.2 Reed-Solomon structure (outer code)

The Reed-Solomon implementation works with byte-sized blocks over a Gallois field GF^8 and thus, it follows the structure $(n, k) = (255, k)$ being k the number of information Bytes. So, the maximum file length that may encode is $255 - (n - k)$ being $n - k$ the redundancy length.

On the other hand, the LDPC-Staircase codec works with symbols multiples of 4 Bytes and the 802.11 frame may have up to 1500 Bytes of length.

For that reason, in order to protect a LDPC-Staircase frame, several Reed-Solomon have to be concatenated as the following figure shows:

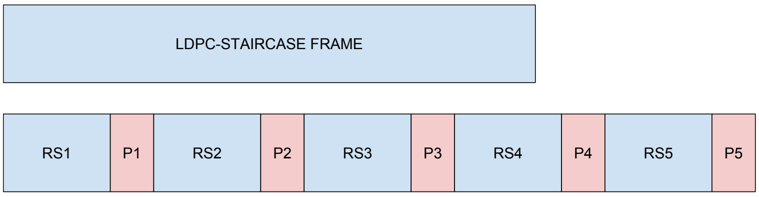


FIGURE 4.8: Reed-Solomon Concatenation.

4.3.3 Interleaver

Reed-Solomon codes are able to correct errors independently of the position inside the Reed-Solomon packet, for that reason interleaving bytes has no benefit.

However, as to protect a LPDC-Staircase packet several Reed-Solomon packets are concatenated, interleaving the whole packet will enhance the recovery capability. That is usually true because in communications channels the errors come in bursts and an interleaver spreads these errors among all the Reed-Solomon packets.

The following figure shows how the interleaver work:

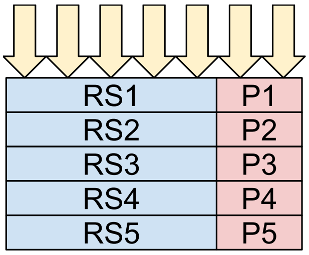


FIGURE 4.9: Packet Interleaving.

The Reed-Solomon packets are placed in rows and then are transmitted by columns. This way the burst errors are spread all over the packet.

4.4 System Implementation Overview

In this section the packet forging, transmission and reception of data will be described step by step to give an overview on how the system works.

4.4.1 Packet Forge and Transmission

The following figure gives an overview on how the system has been implemented:

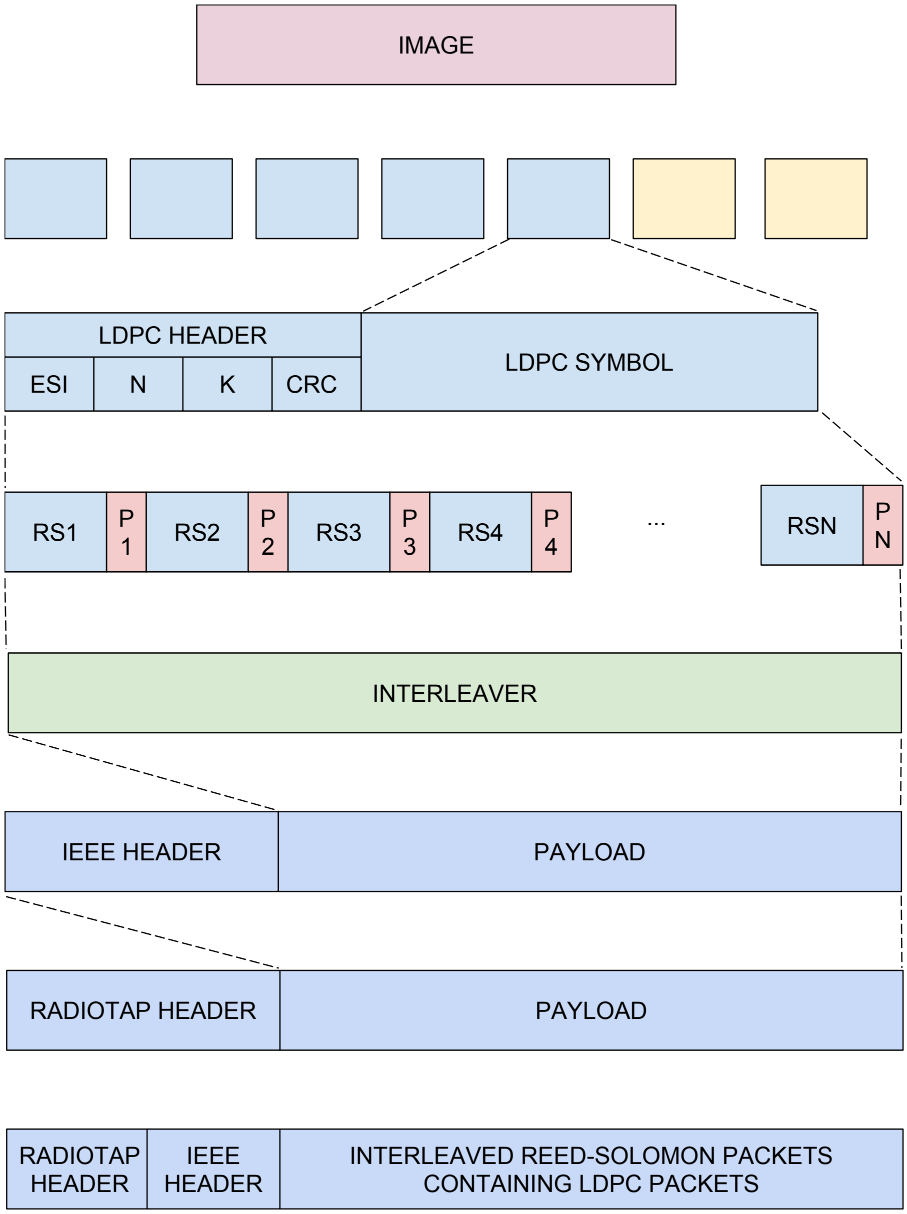


FIGURE 4.10: System implementation.

The forging of the packets goes as follows:

1. The monitor mode of the WiFi adapter is enabled using the iw tool, the bad CRC reception flag is set.

2. A large image is segmented in chunks of length equal to the symbol size of the LPDC-Staircase codec. The symbol size is limited by the 802.3 standard, that defines a maximum packet length of 1500 Bytes. Padding of zeros is applied to the last chunk.

3. The LDPC-Staircase codec generates k source packets and $n-k$ redundant packets according to the specified coding rate, $CodingRate = \frac{k}{n}$ being n the total number of packets (source and redundant).

4. A LPDC-Staircase header that contains a packet identifier, the n and k values and a CRC, is attached in each one of the n packets.

5. As the whole LDPC-Staircase packet is too big to be protected by a Reed-Solomon GF(8) chunk, the LPDC-Staircase is split is several parts smaller than 255 Bytes that are encoded with the Reed-Solomon codec.
6. To maximize the recovery capabilities an interleaver mixes the bytes of the Reed-Solomon concatenated packets.
7. A IEEE 802.11 MAC header with custom addresses is attached to the output of the interleaver.
8. A Radiotap header is attached.

At the end, the frame contains a Radiotap header followed by a IEEE header that has as payload the interleaved concatenation of Reed-Solomon who's payload is the LDPC-Staircase packets. This frame is injected using the *libpcap* library to the Kernel network stack. The Radiotap header will be interpreted by the driver and discarded before transmission. The IEEE header and the rest of the packet will be inserted into the physical layer payload.

4.4.2 Packet Reception and Reconstruction

The reception process follows the inverse path described in figure 4.10

1. The monitor mode of the WiFi adapter is enabled using the *iw* tool.
2. A frame containing a RadioTap Header, IEEE header and payload is received.
3. The Radiotap header data is inspected, treated if necessarily and removed from the frame.
4. The source and destination field of the IEEE header are inspected to determine if the received packet is intended to be delivered to the receiver (this is done by means of a BPF [94], a very optimized packet filter working at the Kernel space). The IEEE CRC field is ignored.
5. The payload of the IEEE frame is deinterleaved and a concatenation of Reed-Solomon packets obtained.
6. The Reed-Solomon chunks are decoded.
7. The CRC field of the LDPC-Staircase packet header is inspected to determine if the Reed-Solomon decoder has successfully decoded the packet. If the CRC check fails the packet is discarded.
8. The n, k and ESI fields of the LPDC-Staircase packet are obtained and passed to the LPDC-Staircase decoder.
9. The LDPC-Staircase decoder decodes the packet. When enough packets are decoded the original image is obtained.

4.5 Conclusion

Chapter 4 has explained the application-layer codification applied to obtain a robust unidirectional link, combating loss-of-signal and bursts errors.

A brief history of error correcting codes has been explained indicating that Reed-Solomon codes are MDS codes with excellent performance against bursts of errors. However, an error correcting code can't combat loss-of-signal.

The robustness of the unidirectional link is achieved by means of loss-resilient codes. Reed-Solomon may also be used as a loss-resilient code, but it has a brutally expensive computational cost. Raptor/Q codes have near-optimal recovery properties, but are computationally impractical for encoding big files. The solution comes from the LDPC-Staircase codes that have low encoding complexity, but high decoding complexity. This way, the complexity of the code resides on reception side where computational power is not a problem, while the encoding is simple and fast for the on-board computer.

In order to have protection against bursts of errors and loss-of-signal a combination of Reed-Solomon and LDPC-Staircase codes is used.

The implemented coding is explained in detail and an overall view of the system is given, describing the each one of the steps of the encoding and decoding processes.

Chapter 5

Experimental Results and Benchmarks

This chapter describes the experimental tests that have been conducted to analyze the performance of the communications system. It starts with the short range test (1.2 km) description and results. Then, it explains how the long range test (188 km) was done and gives an insight on the obtained results.

After that, the chapter explains the system optimization in terms of Reed-Solomon protection and LDPC-Staircase overhead. Finally, a benchmarks section describes the performance of the system in terms of the received power, the Doppler effect and the scintillation effect.

5.1 Short and Long Range Tests

In order to test the system out of the lab environment two tests were performed. The first one was conducted between the near hill located at the north of the UPC campus called Sant Pere Màrtir and the D3 building roof of the UPC. A second long range test of 186 km was performed between Sant Pere Màrtir and Mallorca. Both tests were performed with a Reed-Solomon protection of 35 bytes per packet and coding rates between 0.5 and 0.1.

5.1.1 DataSet and Data Collection

A miscellaneous dataset was prepared to tests the system in different transmission scenarios. It included the following elements:

- Small Text: A poem of Miquel Costa i Llobera in plain text, 2 kB.
- Big Text: Complete book "Don Quijote de la Mancha" in plain text, 320 kB.
- Small Image: Lenna standard test image compressed in jpeg, 790 kB.
- Big Image: ALMA Milky Way image uncompressed tiff, 78 MB.

At the reception side an instance of the reception software was running to demodulate, decode the received packets, and save the obtained results. However, the reception software doesn’t give any feedback to the user about the link status rather than if some packets are being received or not.

In order to obtain the maximum information of every experiment, an instance of the *tcpdump* sniffer was simultaneously launched on background. TCPdump [95] is a sniffer tool that uses *libcap* library. In fact, the development of both projects is linked. When TCPdump is sniffing a wlan interface that is in monitor mode, it grasps additional information directly from the driver by means of the Radiotap Header. From the point of view of the experiments, the relevant information that can be obtained is: the time of arrival of the packets, the RSSI, the packet lengths, the modulation and the FCS field of the MAC layer. All these parameters are obtained per packet basis and permit a detailed study of the channel conditions and system performance.

Besides the dataset, an all-zero file was also sent during the long range test. This file was intended to reduce the time devoted to study the transmission errors. Detecting where the errors have happened once the file has been processed with the LDPC-Staircase code can be tricky because the encoding process is random. However, an all-zero file gives also and all-zero file at the output of the encoder. This way, to determine where an error has been produced it is only necessary to detect where the payload of the packet differs from zero.

A program was written in C to extract the desired information from the data obtained by the *tcpdump* sniffer and a python script was developed to plot the data into graphs using matplotlib [96]. From the dataset and the all-zero files the following graphics were rendered:

- RSSI over time.
- Goodput and Throughput over time.
- CRC fail over time.
- Good vs Bad CRC packets.
- Error length (only for all-zero file).

5.1.2 Short Test: Sant Pere Màrtir - UPC

A short range test was performed between Sant Pere Màrtir and the roof of the D3 building in Campus Nord. The main purpose of this link was to fully test the system and to be sure that there wasn't any bug in the code. It was also the first time that the reception facilities installed for the ^{3}CAT-2 S-band link were tested.

Short Test Specifications

The test was conducted on March 18, 2016 from 12 to 2 PM on a cloudy day. The transmitter was placed at the top of Sant Pere Màrtir hill with quasi-LOS with the D3 building roof. The distance between them was 1.25 km. The default 5 dBi antenna was used at the transmitter side and the 3 m parabolic dish antenna was used on reception. Additionally, a 5 W booster was also tested.

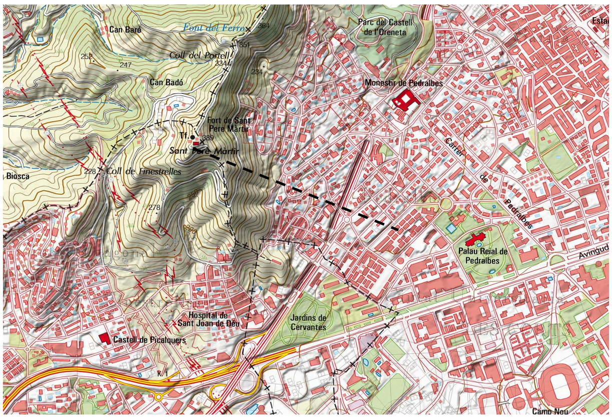


FIGURE 5.1: Topographic map of the short range test, obtained from [97].

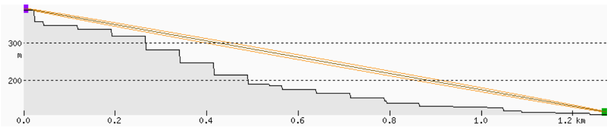


FIGURE 5.2: Elevation profile and first Fresnel zone at 2.4 GHz, calculated with [98].

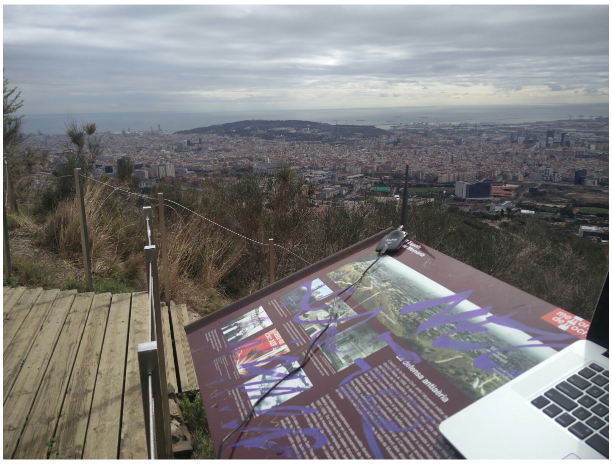


FIGURE 5.3: Panoramic view from the TX side.

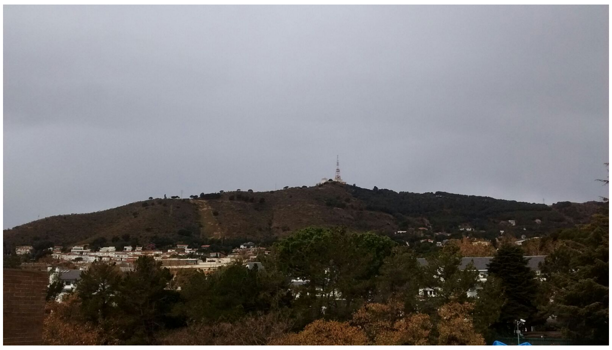
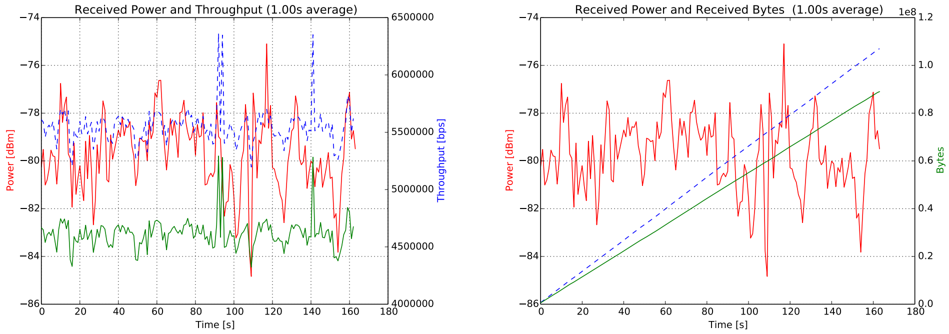


FIGURE 5.4: Panoramic view from the RX side.

Short Range Tests Results

The long image at 6.5 Mbps has been selected to exemplify the performance of the system because it is most similar file to the images that may be captured by the multispectral camera. Similar results were obtained using other samples of the dataset.

The following images show the obtained results in terms of received power and received data:



(A) RSSI in red, Goodput in green and Throughput in blue vs time (1 second average).

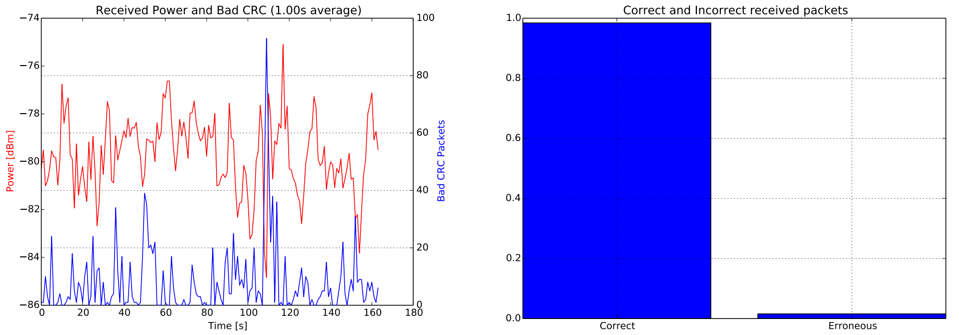
(B) RSSI in red, effective received data in green and total received data in blue vs time (1 second average).

FIGURE 5.5: RSSI, Goodput, Throughput and received data for big image at 6.5 Mbps.

As the figures show, the received power was fluctuating between -78 and -82 dBm. The throughput of the system was pretty stable at 5.6 Mbps, meaning that most of the packets send were received. The B figure shows the accumulated received data vs time. The blue line is the total received data, counting headers and Reed-Solomon protection bytes, whereas the the green line only takes into account image bytes. Both lines have a nearly constant slope as a consequence of the stable reception bit rate. The green line indicates that a 78 MB image was transmitted in less that 150 s.

The bit rate peaks achieving more than 6 Mbps are artifacts produced by erroneous timestamps applied to the received packets by the driver.

The data integrity of the received packets is reflected in the following figures:



(A) RSSI in red, Bad CRC packets in blue vs time (1 second average).

(B) Normalized number of Bad CRC and Good CRC packets.

FIGURE 5.6: Short Range Test: Error analysis for big image at 6.5 Mbps.

It is clearly visible how there is a correlation between received power and erroneous packets. When the received power goes below -85 dBm there is a peak on the received packets that contain errors. However, the number of packets with CRC fail is really low compared with the number of packets received without errors.

The results were promising and showed that the system was working appropriately, being able to recover the send files even when there were loss of packets.

5.1.3 Long Range Test: Sant Pere Màrtir - Serra de Tramuntana

After being completed the short test test with successful results, demonstrating that the system was able to perform as expected, a long range test was planned. Long range test are sometimes difficult to perform due to the Earth's curvature that impedes the line-of-sight between distance points.

Taking advantage that Barcelona is a city beside the Mediterranean Sea and that the author is original from Mallorca (which is visible from Barcelona) a long range link between Barcelona and Mallorca was proposed.

5.1.4 Long Test Specifications

Barcelona and Mallorca are separated by roughly 190 km, which is the required elevation for the transmitter and receiver to have line-of-sight? The following scheme shows the trigonometry problem:

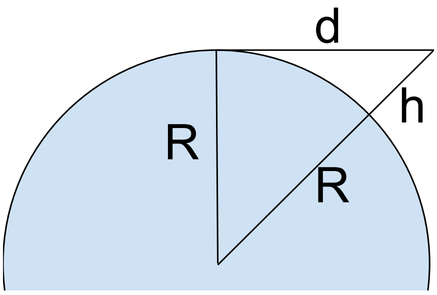


FIGURE 5.7: Earth Horizon Scheme.

Applying the Pitagoras theorem we have that:

$$d = \sqrt{(R+h)^2 - R^2}. \tag{42}$$

As the radius of the Earth is much more large the the altitude, $R >> h$ then the formula can be simplified as:

$$d = \sqrt{R^2 + h^2 + 2Rh - R^2} = \sqrt{h^2 + 2Rh} \approx \sqrt{2Rh}. \tag{43}$$

For 190 km of distance and considering that both side to be at the same elevation the transceiver should be placed at:

$$h = \frac{90^2}{2 \cdot 6370} = 635\,\mathrm{m}. \tag{44}$$

The highest point in Barcelona is located at the Tibidabo mountain, with 512 meters above the sea level. However, there is an amusement park at the top and the access is limited. A more convenient location is the top of the Sant Pere Màrtir hill, at 370m above the sea level. From this point, the horizon is located at:

$$d \approx \sqrt{2 \cdot 6370000 \cdot 370} \approx 69\,\mathrm{km}. \tag{45}$$

Knowing where one antenna is going to be placed and how much distance it covers, the altitude in which the second antenna can be calculated:

$$190 - 69 = \sqrt{2 \cdot R \cdot h}, \tag{46}$$

$$h = \frac{121000^2}{2 \cdot 6370000} \approx 1150\,\mathrm{m}. \tag{47}$$

The highest peaks in Mallorca are located in the Serra de Tramuntana, a mountain range running SouthWest–NorthEast of the island. Its major peak is Puig major with 1440 m followed by Massanella, which has 1360 m. The antenna has to be placed in a position that had no obstacles towards Barcelona and with sufficient altitude. Both conditions reduce the potential locations to a really reduced number.

In first instance, the public roads where inspected looking for some point where the requirements were meet. However, there are only two roads in Mallorca above 1000 meters having direct visibility towards Barcelona. Unfortunately, they are closed roads being one of private use and the other one part of a military area. After unsuccessful attempts to obtain access to these roads the only solution was to do some trekking.

After a detailed study of the oreography and as a tradeoff between elevation and access difficulty the following location was selected:

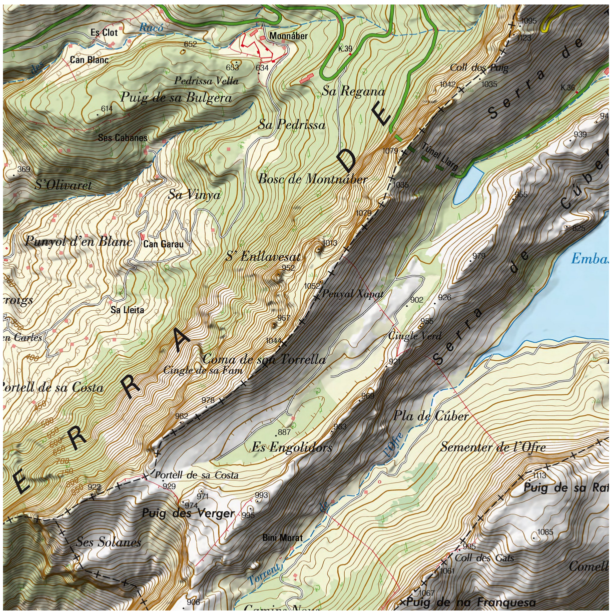


FIGURE 5.8: Location at Serra de Tramuntana, Penyal Xapat, image from [97].

The highest point of the Ma-10 public road in Mallorca (km 37) was taken as an advantage to start the trekking from the highest possible altitude. From that starting point and walking a 40-minute off-road track, it is possible to reach the Penyal Xapat peak at 1050 meters of altitude, from which there are no obstacles pointing to Barcelona.

Although from this altitude there is not straight line-of-sight to Sant Pere Màrtir the effect of the refraction has not been considered in the equations. Fortunately, the refraction effect makes the Earth's curvature apparently flatter. An approximation of the refraction effects may be accomplished by multiplying the radius of the Earth by the factor 7/6 [99] giving the following result:

$$d_{SantPereMrtir} = \sqrt{2 \cdot \frac{7}{6} \cdot 6370000 \cdot 370} \approx 74km \tag{48}$$

$$d_{SerraTramuntana} = \sqrt{2 \cdot \frac{7}{6} \cdot 6370000 \cdot 1050} \approx 125km \tag{49}$$

Thus, the total distance would be $74 + 125 = 199$ km which covers the 188 km of distance between the two selected points meaning that there is virtually line-of-sight. In fact, the following image depicts how the line of sight and half of the first Fresnel zone are compromised:

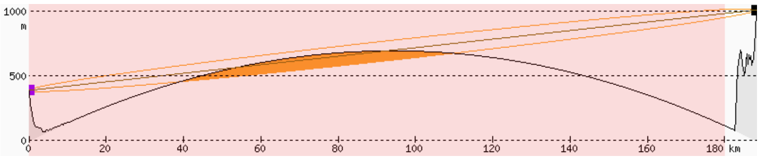


FIGURE 5.9: Elevation profile and first Fresnel zone at 2.4 GHz, image from [98].

It is worth noting that the simulation software it is not taking into account the refraction effects.

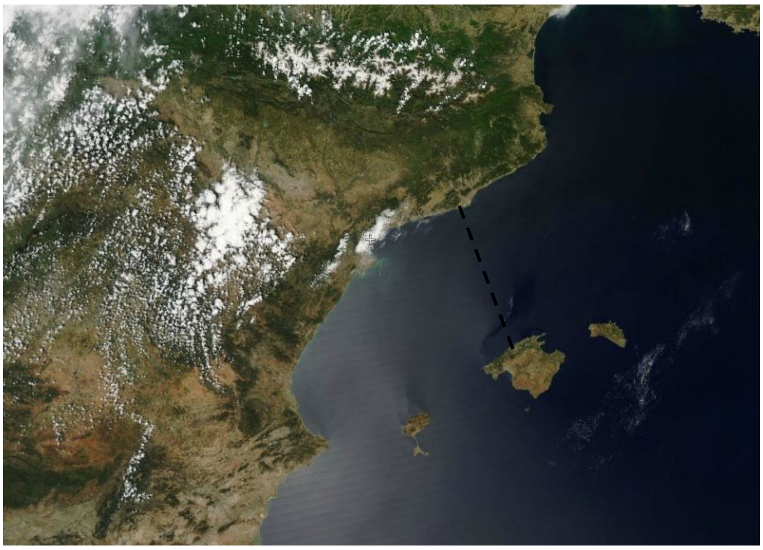


FIGURE 5.10: Long range link map, image obtained from [100].

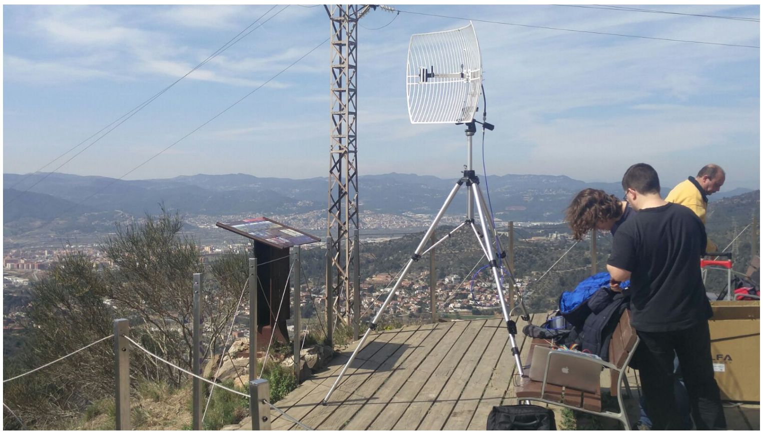


FIGURE 5.11: Transmitting antenna, Sant Pere Màrtir, Barcelona, 370 m.

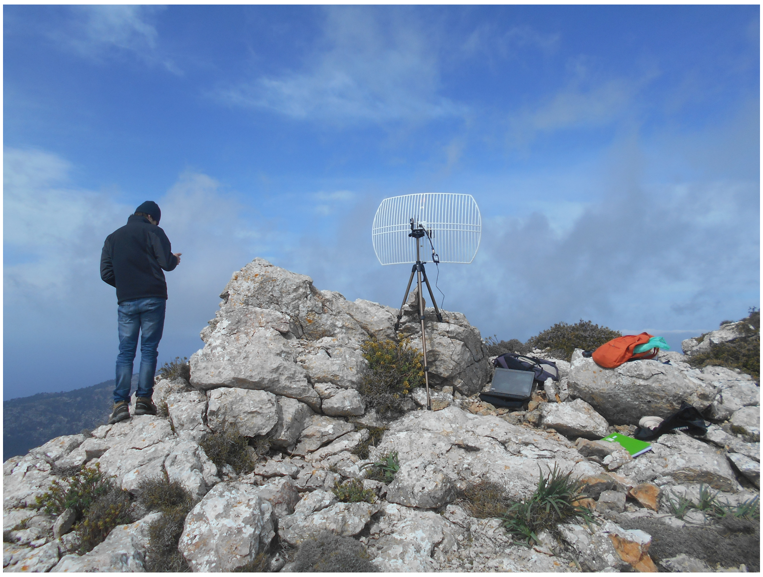


FIGURE 5.12: Transmitting antenna, Penyal Xapat, Serra de Tramuntana, Mallorca, 1050 m.

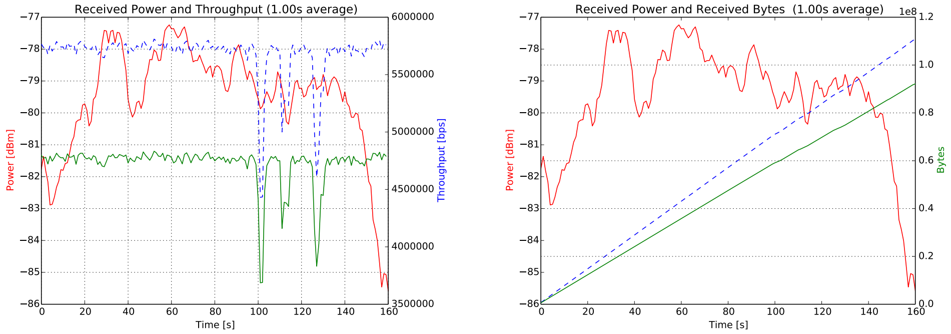
In order to cope with the path loss two 24 dBi parabolic antennas [101] were purchased and used in both transmitter and reception sides. The test was performed the 23rd of March from 12 PM to 2.15 PM. It was sunny day in Barcelona, but in Serra de Tramuntana it was a cold day, the sky was cloudy, the visibility was reduced to some meters and there were sustained heavy winds. The weather conditions greatly difficulted the antenna pointing in reception.

The two teams established voice and data communications using mobile networks to coordinate the tests that last until the battery of the transmitting side laptop was completely drained.

Long Range Test Results

The long range tests results were similar to the ones obtained during the short range tests, although in this case the received power had much bigger fluctuations due to the vibrations induced by wind at the reception antenna. However, the performance was outstanding taking into account the 188Km of separation between transmitter and receiver demonstrating the the system performance.

As in the short range tests, the transmission of the large image has been used to exemplify the performance of the system:



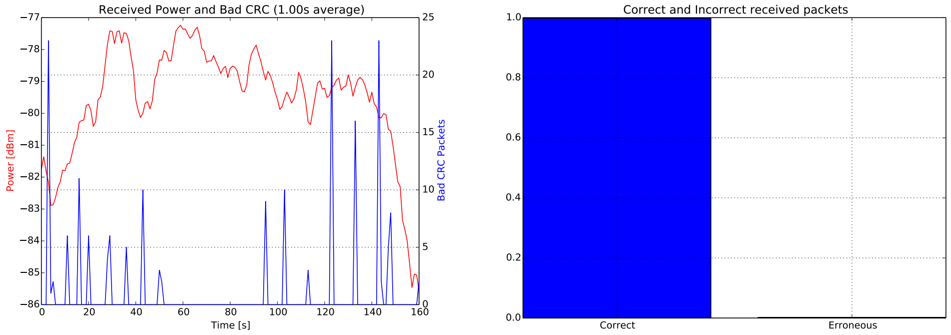
(A) RSSI in red, Goodput in green and Throughput in blue vs time (1 second average).

(B) RSSI in red, effective received data in green and total received data in blue vs time (1 second average).

FIGURE 5.13: Long Range Test: RSSI, Goodput, throughput and received data for big image at 6.5 Mbps.

As the images show, the received power had severe fluctuations, however, the signal strength was quite good taking into account that the link was established at 188 km of distance. The throughput was quite constant at 5.6 Mbps meaning that a 78 MB image was transmitted in less than 2 and a half minutes. This results are similar to the ones obtained in the short range test. Even though, abrupt decays of bit rate were registered as the one that happens after second 100 due to instantaneous loss of packets. However, as the received power was several dBs over the sensitivity level, most of the packets were received correctly and the bit rate was quite stable. The graph on the right shows how the accumulated data was fairly constant over the test.

The following two images describe show the data integrity:



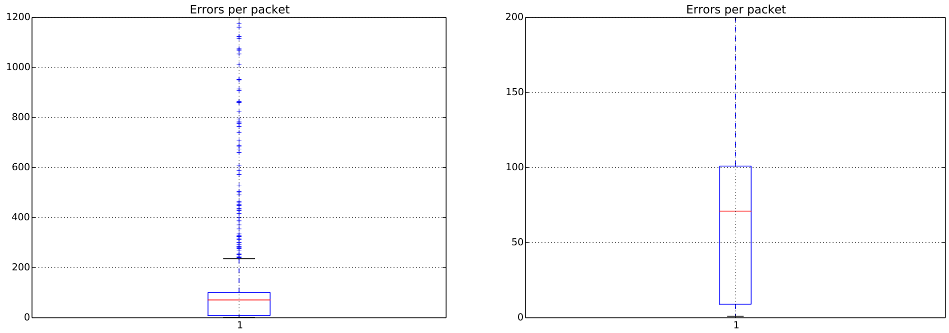
(A) RSSI in red, Bad CRC packets in blue vs time (1 second average).

(B) Normalized number of Bad CRC and Good CRC packets.

FIGURE 5.14: Long Range Test: Error analysis for big image at 6.5 Mbps.

The results of the data integrity are very similar to the ones obtained in the short range tests, being the number of packets received with errors also minimal. In fact, the average received power was higher than in the short range, and the increase in SNR lead to a lower number of bad CRCs. In conditions where the received power is well above the receiver sensitivity the Reed-Solomon protection is useless, because almost all the packets are received without errors, and it reduces the goodput introducing an overhead in every packet.

In order to determine the nature of the received errors, an all-zero file was send at 1 Mbps. A posterior study of the detected errors can be seen in the following images:



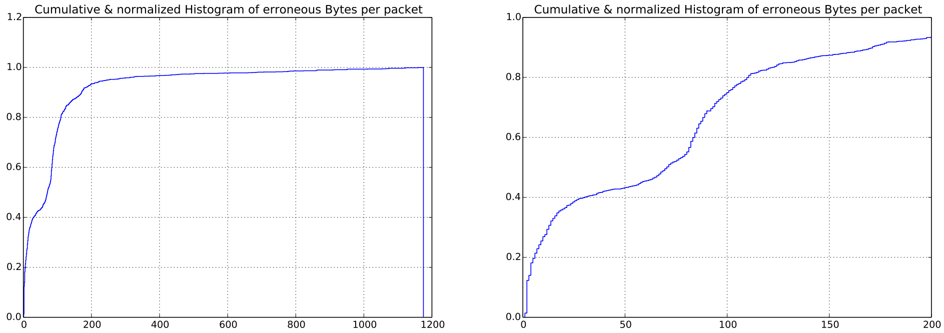
(A) Boxplot of erroneous bytes per packet.

(B) Boxplot of erroneous bytes per packet (zoom).

FIGURE 5.15: Long Range Test: Box plots of error lengths for big image at 1 Mbps.

The boxplot representation is useful to determine the stadistics of the errors. It can be seen how most of the errors have a length below 100 Bytes per packet and the median of erroneous bytes per packet is around 70.

The second representation, which is a cumulative and normalized histogram probably gives a better insight on the error lengths:

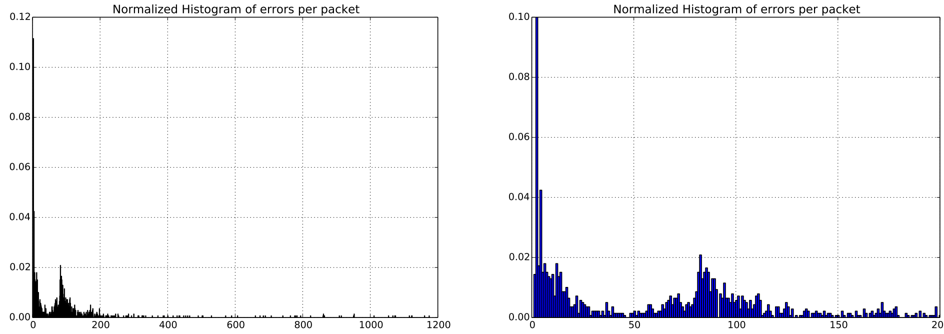


(A) Histogram of byte errors per packet.

(B) Histogram of byte errors per packet (zoom).

FIGURE 5.16: Long Range Test: Cumulative and normalized histograms of error length for big image at 1 Mbps.

A relevant result that these figures show is that packets with a few errors have high frequency. For example, the packets with less than 40 errors represent more than the 40% of the total packet with errors. In the following non-cumulative histograms it is clearly visible the peak located at the left part of the graphics, indicating a high frequency of occurrence of packets that have a few errors.



(A) Normalized Histogram of byte errors per packet.

(B) Normalized Histogram of byte errors per packet (zoom).

FIGURE 5.17: Long Range Test: Histograms of byte errors for big image at 1 Mbps.

5.2 Optimization

The results obtained in both short and long range tests were satisfactory. However, most of the parameters of the system were set as best-guests and without having conducted any rigorous study.

5.2.1 Reed-Solomon Optimization

The Reed-Solomon is applied as outer code of the LDPC-Staircase packets. The application of the Reed-Solomon codes will be beneficial if the following simple condition applies:

$$RepairedBytes > OverheadBytes_{GoodCRCPackets}. \tag{50}$$

That is, the number of repaired bytes has to be greater than the overhead on the packets without errors created by the parity bytes. The former expression may be written as:

$$\begin{gathered} BadCRCs \cdot RecoverableFraction \cdot PacketSize \\ > \\ GoodCRCs \cdot Overhead. \end{gathered} \tag{51}$$

Being the recoverable fraction the number of packets in which the number of errors is equal or less the the recovery capability of the Reed-Solomon codes. It is worth noting that a perfect spreading of the errors by the interleaver is being considered. If the errors are not perfectly spread the recovery performance of the Reed-Solomon would decrease.

The optimum number of Reed-Solomon parity bytes will depend on the length of the errors and the fraction of packets received with errors that. At the end, all the parameters depend directly or indirectly on the instantaneous received power.

To test the system with different scenarios of received power a USRP SDR from Ettus [102] was used to simulate the desired channels conditions. The following figure describes the followed steps:

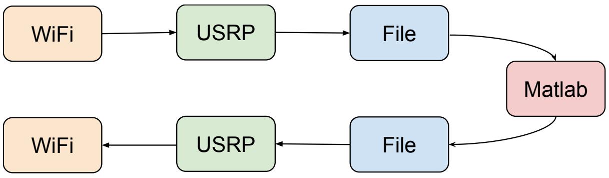
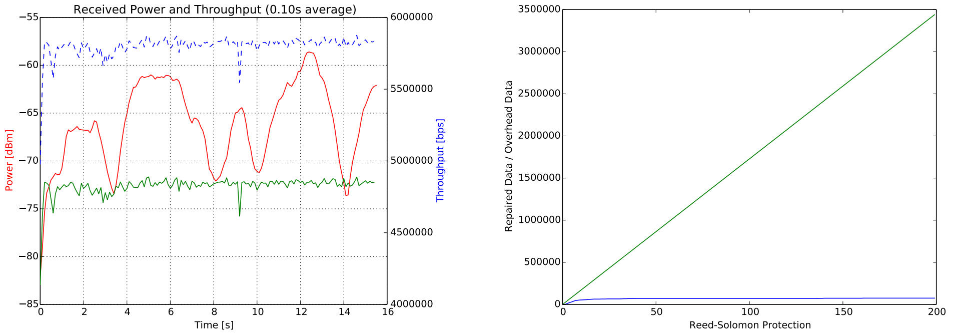


FIGURE 5.18: Channel simulation scheme.

A WiFi adapter transmitted data and the USRP SDR saved the I/Q components into a file. This file was modified with Matlab applying the desired channel mode. In order to simulate worst-case conditions a scintillation of $S_4 = 1$ was chosen. Finally, the modified I/Q file was fed into the SDR that transmitted it and was received by a WiFi adapter.

Once the simulated channel was obtained several transmissions were done changing the average transmitted power and saving the results with *tcpdump*. The errors of the received packets were analyzed in order to determine which would have been the performance of a Reed-Solomon code depending on the number of parity bytes.

The following images show the performance when the received power is at 15 dB above optimal performance sensitivity level:



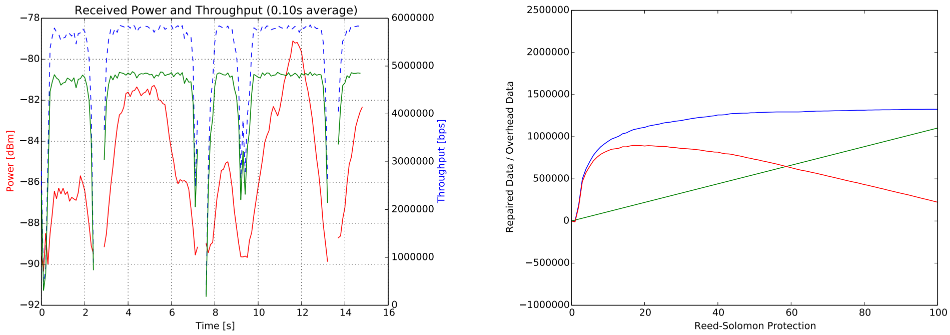
(A) RSSI in red, Bad CRC packets in blue vs time (0.1 second average).

(B) Reed-Solomon recovered bytes in blue and overhead in green.

FIGURE 5.19: Reed-Solomon optimization: Average received power 15 dB above optimal performance sensitivity level.

In those conditions only 0.5% of packets had errors and the overhead introduced by the Reed-Solomon was much greater than the benefits of using it. Thus, when the link is working with great margin over the optimal performance sensitivity level, it is better not to use Reed-Solomon.

However, as the link is expected to work under almost no margin, the performance of the system was studied when the received power is close to the optimal performance sensitivity level at 6.5 Mbps:



(A) RSSI in red, Bad CRC packets in blue vs time (0.1 second average).

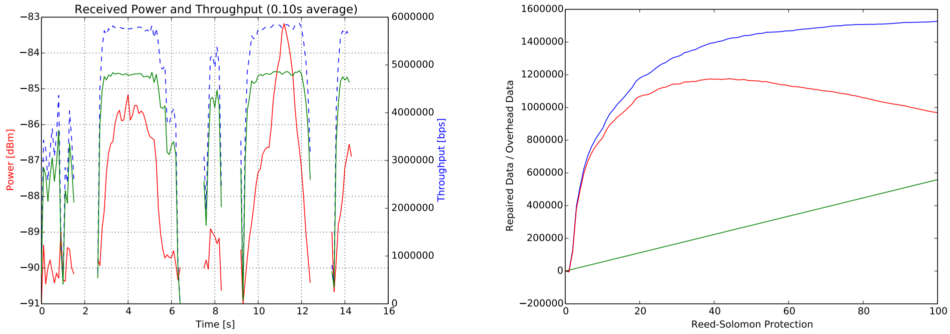
(B) Recovered bytes in blue and overhead in green, difference in red.

FIGURE 5.20: Reed-Solomon optimization: Average received power 0 dB above sensitivity level.

The left image shows how there were loss-of-signal produced by deep fadings, in fact 18% of packets were received with errors. The right image shows the overhead and recovery curves and its difference. As the equation 50 indicates, when the recovered bytes are above the overhead bytes it is beneficial to use Reed-Solomon. As the right image shows, that happens for Reed-Solomon recovery capability between 3 and 100. However, the optimal Reed-Solomon recovery length is given by the point where the difference between the recovered bytes and overhead is maximum and positive. As the red curve is the difference between the recovered bytes and the overhead, the optimal recovery performance is located when the red curve is positive and has a maximum.

In that case, the optimal value is a Reed-Solomon recovery capability of 18 over the total packet length, meaning that 36 recovery bytes have to be split between all the Reed-Solomon blocks.

In order to know which would be optimum Reed-Solomon size where the channel conditions are even worse, the system performance at 3dB below the sensitivity level was studied:



(A) RSSI in red, Bad CRC packets in blue vs time (0.1 second average).

(B) Recovered bytes in blue and overhead in green, difference in red.

FIGURE 5.21: Reed-Solomon optimization: Average received power -3 dB above sensitivity level.

In that case 34% of packets were received with errors. The left image shows abundant loss-of-signal and the right says that the optimum Reed-Solomon protection is around 45. Thus, the total number of parity bytes to be split between the Reed-Solomon blocks should be of 90.

Lower values of average received power have not been considered as they would turn out to be insufficient for transmitting all the desired data with a single satellite pass-by. It has been seen that the lower is the received power the higher is the optimum Reed-Solomon protection. However, the optimum recovery lengths are pretty small compared with the total packet length, meaning that it is better to focus on recovering packets with a few errors rather than trying to repair packets with a high number of errors that would cause a lot of overhead.

5.2.2 LDPC-Staircase Optimization and Testing

5.2.3 Encoding Speed

The encoding speed of the LDPC-Staircase codec is really fast. The codec has been tested on a Raspberry A+ 700 MHz of clock and 256 MB of RAM and on a Raspberry B+ featuring the same clock, but 512 MB of RAM. The following table contains the encoding speed for a 78MB image and various coding rates:

TABLE 5.1: LDPC-Staircase codec encoding speed.

Coding Rate	Raspberry A+	Raspberry B+
$r = 0.75$	45 s	8 s
$r = 0.5$	110 s	10 s
$r = 0.25$	-	19 s

The Raspberry A+ encoding speed is highly reduced due to its scarce RAM, not even being able to encode the image with $r = 0.25$. The Raspberry B+, having the same processor but 512MB of RAM has an impressive performance in encoding speed.

5.2.4 Decoding

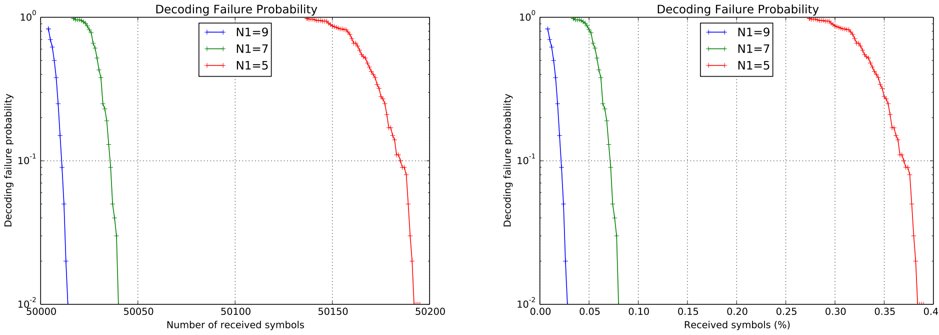
Overhead

The overhead is the additional amount of data that needs to be received over the original file size. The OpenFEC LDPC-Staircase codec features two kinds of decodification algorithms: Iterative decoding and Maximum-Likelihood decoding, both algorithms are described in the main author's Thesis [103]. Basically, the iterative decoding algorithm is really fast but it requires a higher ϵ and the maximum likelihood algorithm is capable to decode packets with low ϵ at the expense of a much higher computation time.

The overhead of the LDPC-Staircase codec may be controlled adjusting the N1 parameter of the codec. This parameter describes how many ones per column are in the left side of the parity check matrix. Adjusting the

N1 it is possible to balance the tradeoff between overhead and decoding complexity.

The following figures show the decoding failure probability as a function of the number of the number of received packets for a coding rate of 0.75:



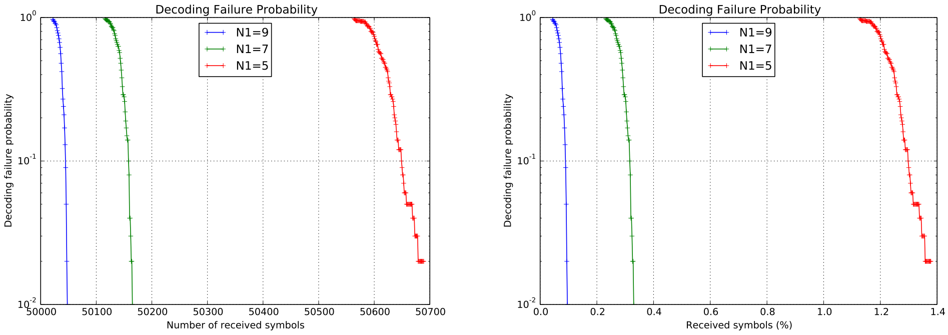
(A) Decoding failure probability as a function of the number of received packets for 0.75 coding rate.

(B) Decoding failure probability as a function of the amount of received data in % for 0.75 coding rate.

FIGURE 5.22: Decoding failure probability as a function of received data for 0.75 coding rate.

The figures show how the LDPC-Staircase codes overhead behaves similar to an ideal fountain code approaching zero. The highest is the N1 value, the lower may be the amount of data recovered data to obtain the original file. For N1=9, and 50000 source symbols, with only 20 extra symbols received the probability of decoding failure is lower than 10^{-2}, this is only a 0.0025% of overhead!

When the coding rate is lowered (more repair symbols are created) the overhead slightly increases:



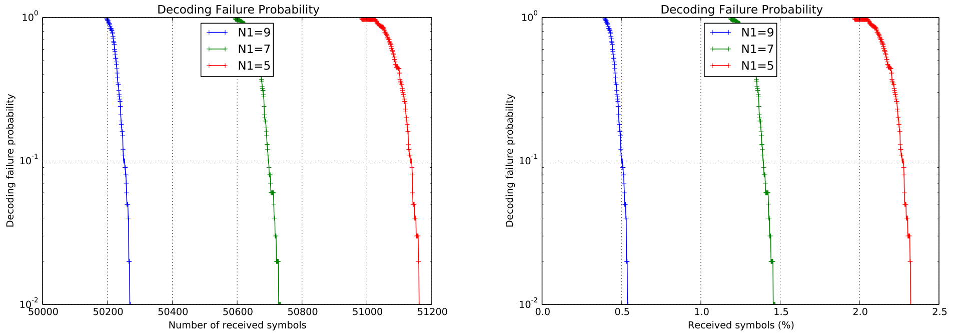
(A) Decoding failure probability as a function of the number of received packets for 0.5 coding rate.

(B) Decoding failure probability as a function of the amount of received data in % for 0.5 coding rate.

FIGURE 5.23: Decoding failure probability as a function of received data for 0.5 coding rate.

Moving from a coding rate of 0.75 to 0.5 has approximately triplicated the amount of data to be received. However, as the pictures show, the performance continues being really good.

Finally, the recovery overhead of the LDPC-Staircase codes was testes for a coding rate of 0.25 giving the following results:



(A) Decoding failure probability as a function of the number of received packets for 0.25 coding rate.

(B) Decoding failure probability as a function of the amount of received data in % for 0.25 coding rate.

FIGURE 5.24: Decoding failure probability as a function of received data for 0.25 coding rate.

It is clearly visible how the overhead is much greater than in the two former code rates. However, even in this extreme case the code continues exhibiting a great recovery performance.

Memory Consumption

The memory consumption when encoding is approximately given by:

$$EncodingMemory \approx FileSize + FileSize \cdot \left(\frac{1}{CodingRate} \right) \tag{52}$$

The encoding speed will be high if the RAM is higher than the Encoding Memory. For this reason, at least 512 MB of RAM are advised.

The decoding used memory will depend on the decoding method used. For Iterative decoding the memory usage is equal to the encoding memory. However, for the Maximum-Likelihood decoding the required memory increases rapidly. The following table summarizes the required memory for doing a ML decoding over a 80 MB file:

TABLE 5.2: Memory consumption for 80 MB and distinct code rates.

Rate	Memory
$r = 0.8$	190 MB
$r = 0.7$	290 MB
$r = 0.6$	550 MB
$r = 0.5$	1000 MB
$r = 0.4$	1800 MB

As the table shows, when the Maximum Likelihood decodification is applied the required RAM memory skyrockets. However, to consider really low values of coding rate has no sense because the satellite wouldn't have time to transmit all the data during a pass-by. For moderate values of coding rate, the memory requirements are perfectly compatible with nowadays consumer-grade computer specifications. As an extreme use-case, for a coding rate $r = 0.2$ the expected memory usage to decode a 100 MB file would be less than 8 GB of RAM, which is also perfectly affordable value.

5.3 Benchmarks

5.3.1 Performance vs Received Power

RSSI indicator vs Real Received Power

The *libpcap* library fetches directly from the driver the RSSI values. The received power in dBm is calculated by the adapter using an indirect method by inspecting the physical preamble of every packet, further details may be seen in the Atheros patent [104].

As this indirect method is prone to suffer errors a series of tests have been conducted to determine the exactitude of the readings. An adapter was injecting packets and another adapter was receiving them. In between, the channel attenuation was controlled by means of RF attenuators from Minicircuits [105]. Transmitter and receiver were separated by 4 meters of coaxial cable [106], which introduced 3 dB of extra loss.

As an example, the characterization of RSSI versus real received power for 6.5 Mbps is shown. The Attenuation starts at 98 dB and ends at 112 dB, the step is 1 dB:

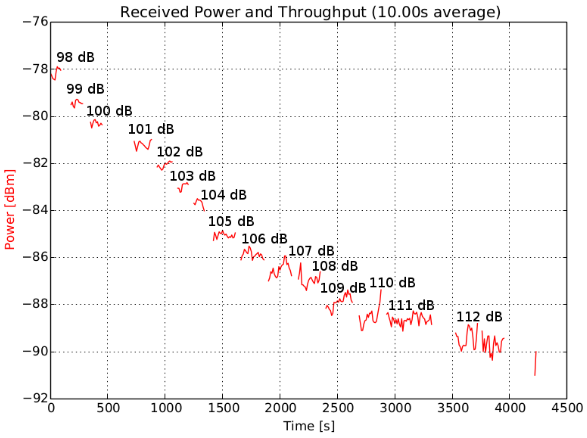


FIGURE 5.25: RSSI vs real attenuation for 6.5 Mbps.

As the increase in attenuation is linear if the RSSI was perfect it would have to be linear too. The figure shows that at the beginning, when the received signal is high, the linearity is maintained. However, for low reception power values the linearity is partially lost. As the transmitted power of the adapter is 21 dBm (measured at 3.8.1) it is possible to conclude that the RSSI indicated by the adapter is usually around 1-2 dB below the real level. The same results were obtained for other modulations.

Received Power vs Bit Rate

In order to determine the performance of the system in terms of the received power several tests on received power vs bit rate achieved were done. To exemplify the obtained results the following image shows a realization for 6.5 Mbps:

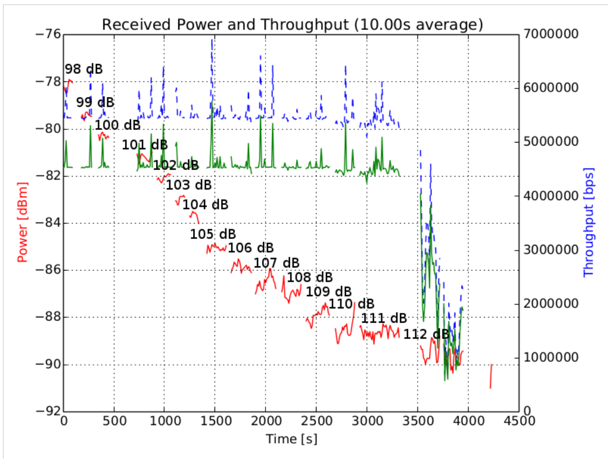


FIGURE 5.26: RSSI vs bit rate test for 6.5 Mbps.

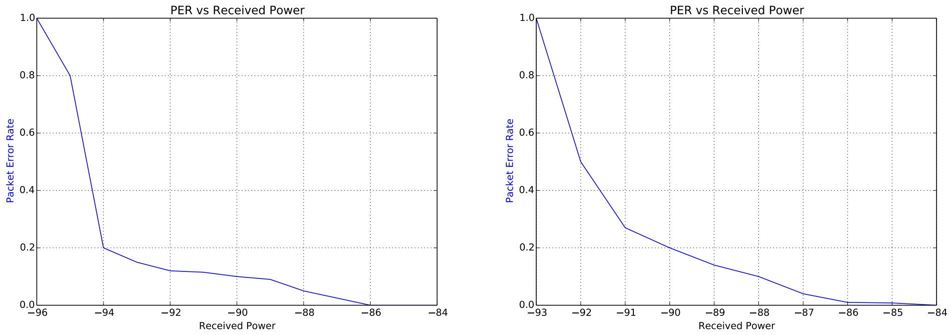
As it can be seen the bit rate is maintained constant until the received power reaches a level were the system collapses and there is a drop in the received packets rate. The minimum received power for optimal operation in terms of received bit rate was calculated for several modulations:

Modulation	Min dBm for optimum bit rate
1 Mbps CCK	-93
2 Mbps CCK	-90
6.5 Mbps OFDM	-86

TABLE 5.3: Minimum reception power for optimum bit rate performance.

Received Power vs Packet Error Rate

In order to determine how the WiFi adapter behaves with distinct values of received power, the data obtained in all the simulations has been used to determine the relation between received power and packet error rate. The higher and lower bit rate modulations were selected obtaining the following curves:



(A) PER as a function of received power for 1 Mbps.

(B) PER as a function of received power for 6.5 Mbps.

FIGURE 5.27: PER vs received power.

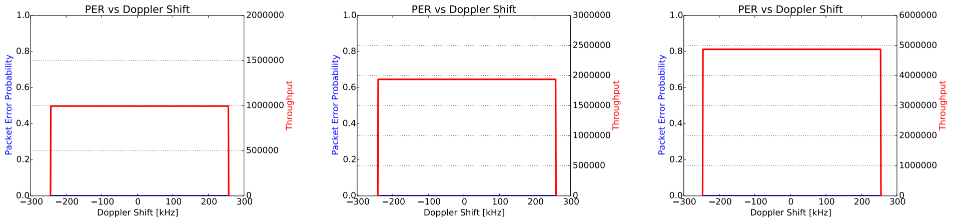
As expected, the 1 Mbps modulation is more robust performing around 4 dB better than the 6.5 Mbps modulation for the same value of received power. Even though the performance in terms of PER of the 6.5 Mbps curve seems to be pretty good it has to be compared with the former figure 5.26, in which it can be seen how the bit rate decreases drastically when received power is below -86 dBm. For that reason, the target received power when using 6.5 Mbps modulation should be greater than -86 dBm.

5.3.2 Doppler

As stated in section 2.3.3 the maximum Doppler effect will be of around ±62 kHz. In this section the frequency shift tolerance of the WiFi adapters will be studied. The USRP SDR was used to save raw I/Q data modulated by a WiFi transceiver. Afterwards the file was transmitted changing the central frequency with the desired shift.

DSSS modulations

The three tested DSSS modulations (1,2 and 5.5 Mbps) turn out to be robust against Doppler shifts. The system collapses when the applied shift is greater than ±250 kHz:



(A) Throughput and packet errors vs Doppler shift for 1 Mbps.

(B) Throughput and packet errors vs Doppler shift for 2 Mbps.

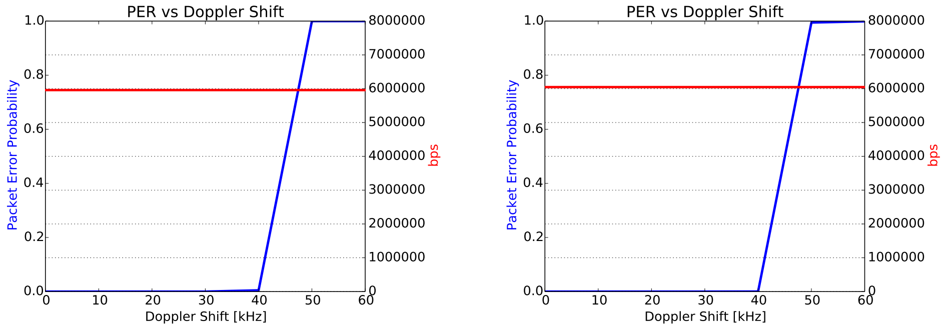
(C) Throughput and packet errors vs Doppler shift for 5.5 Mbps.

FIGURE 5.28: Throughput and packet errors vs Doppler shift for DSSS modulations.

Using DSSS modulations the demodulator wouldn't have problems with the Doppler effect because the maximum shift is supported by the adapters with great margin.

OFDM modulations

The multicarrier nature of OFDM modulations makes them much vulnerable to slight changes in frequency. As the carriers are separated by 312.5 kHz even a small change in frequency will produce inter-carrier interference. For both OFDM tested modulations (6 Mbps with 48 subcarriers and 6.5 Mbps with 52 subcarriers) the results were pretty much the same:



(A) Throughput and packet errors vs Doppler shift for 6 Mbps.

(B) Throughput and packet errors vs Doppler shift for 6.5 Mbps.

FIGURE 5.29: Throughput and packet errors vs Doppler shift for DSSS modulations.

The image shows how there are errors starting from 40 kHz of Doppler shift that rapidly increase towards a PER=1. However, the bit rate continues being constant. Using OFDM modulation the WiFi adapter in reception would experience problems for high values of Doppler shift.

The simplest solution to solve the Doppler effect is to only use DSSS modulations.

5.3.3 Scintillation effects

The scintillation effect will produce fadings and phase changes that will corrupt the packets and produce loss-of-signal. To combat these effects the LDPC-Staircase code has been implemented. However, one of the key parameters of this codes is the coding rate, which has to be adapted to the channel conditions. On the one hand, a too low coding rate will generate a lot of repair packets and would be a waste of resources. On the other hand, a too high coding rate may not be sufficient to deal with the packet losses.

Several tests have been conducted in order to know which is the effect of the phase changes and fadings over the packet errors and losses. Both effects have been studied separately:

5.3.4 Packet loss vs Scintillation τ_0 index

In order to study the effects of the τ_0 parameter without being affected by the intensity parameter the mean received power level was set to a value in which the fadings didn't surpass the receiver sensitivity.

The obtained results for low levels of τ_0, which mean rapid changes in phase, show that the WiFi adapters are able to track the phase changes and the induced errors are negligible.

The next image shows a channel simulation with a really low value of $\tau_0 = 0.01$ s which introduces rapid phase changes. The arrows point to the most extreme phase shifts that the scintillation has introduced.

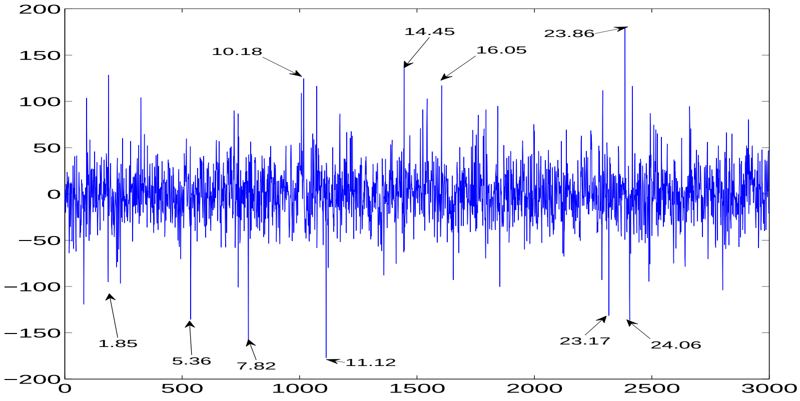


FIGURE 5.30: Example of phase changes for $\tau_0 = 0.01$ s with markers indicating maximum phase peaks.

In reception, even in these tricky conditions, the PLLs are able to follow the phase changes and the received errors were minimal. In this realization 31 errors were detected, 8 of these occurred exactly at the same position the markers are pointing to. The following table summarized the time of arrival of packets with errors compared with the positions of the maximum phase shifts.

Occurrence of Errors (s)	Marker Positions (s)
1.847969	1.85
5.368348	5.36
10.178349	10.18
11.118111	11.12
14.449354	14.45
23.169488	23.17
24.057988	24.06

TABLE 5.4: Occurrence of errors vs maximum phase changes.

This results highlights the direct correlation between high phase changes and received errors. However, as stated, the number phase induced errors are absolutely negligible.

5.3.5 Packet loss vs Scintillation S_4 index

It has been seen how the phase changes will have minimum impact over the packet loss. However, on a thigh power margin the S_4 index will have a strong impact on the number of packets that will be lost.

In order to determine the impact of the fadings induced by the scintillation effects packets received with signal strength below the sensitivity have been considered as lost packets. The packet loss has been calculated for different values of S_4 and received power margin over sensitivity level.

The following graph summarizes in a compact fashion the obtained results.

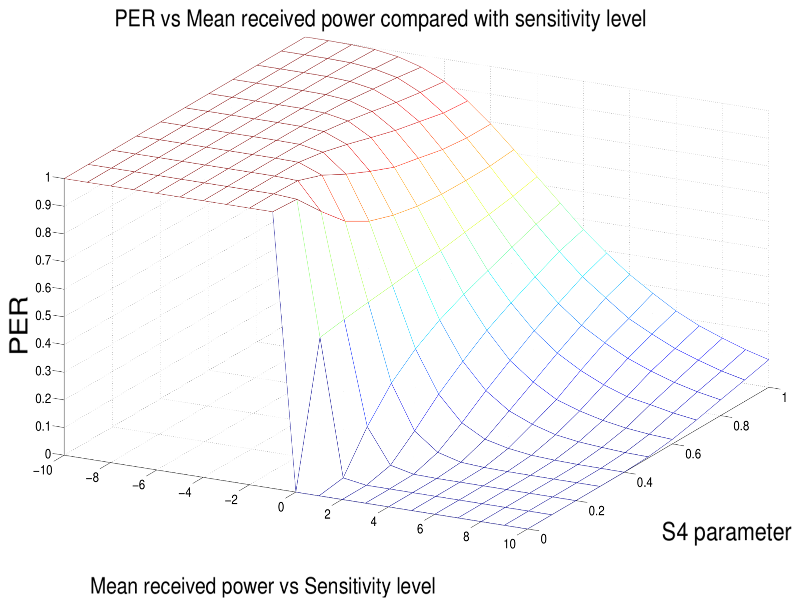


FIGURE 5.31: PER as a function of the S_4 index and received power compared to the sensitivity level.

Using this data it is possible to select the appropriate coding rate to be applied by the LPDC-Staircase codec, by means of estimating the S_4 parameter (that will depend on the Earth's region), and the link budged.

	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
+10dB	0	0	0	0	0	0	0.02	0.04	0.06	0.08	0.1
+9dB	0	0	0	0	0	0.01	0.02	0.05	0.07	0.1	0.12
+8dB	0	0	0	0	0	0.01	0.03	0.06	0.09	0.12	0.15
+7dB	0	0	0	0	0	0.02	0.05	0.08	0.12	0.15	0.18
+6dB	0	0	0	0	0.01	0.03	0.07	0.11	0.15	0.19	0.22
+5dB	0	0	0	0	0.02	0.06	0.09	0.15	0.2	0.23	0.27
+4dB	0	0	0	0.01	0.04	0.09	0.14	0.2	0.25	0.29	0.33
+3dB	0	0	0	0.03	0.09	0.15	0.21	0.26	0.31	0.36	0.39
+2dB	0	0	0.02	0.09	0.18	0.24	0.3	0.35	0.39	0.44	0.46
+1dB	0	0.01	0.15	0.25	0.33	0.38	0.42	0.46	0.49	0.52	0.55
+0dB	0	0.51	0.52	0.53	0.54	0.55	0.57	0.58	0.59	0.61	0.63
-1dB	1	0.99	0.9	0.81	0.76	0.73	0.71	0.7	0.7	0.7	0.71
-2dB	1	1	1	0.97	0.92	0.88	0.84	0.82	0.8	0.79	0.79
-3dB	1	1	1	1	0.98	0.96	0.93	0.91	0.89	0.87	0.86
-4dB	1	1	1	1	1	0.99	0.98	0.96	0.95	0.93	0.92
-5dB	1	1	1	1	1	1	1	0.99	0.98	0.97	0.96
-6dB	1	1	1	1	1	1	1	1	1	0.99	0.98
-7dB	1	1	1	1	1	1	1	1	1	1	0.99
-8dB	1	1	1	1	1	1	1	1	1	1	1
-9dB	1	1	1	1	1	1	1	1	1	1	1
-10dB	1	1	1	1	1	1	1	1	1	1	1

TABLE 5.5: PER as a function of the S_4 index and received power/sensitivity level difference table.

5.4 Conclusion

Chapter 5 has described the short and long range tests. It has also given the results obtained in both tests and during the optimization process.

The short range test demonstrated the viability of the system. It confirmed that the system was able to reliably transmit data over a unidirectional link with packet losses. A fairly constant throughput of 5.6 Mbps was achieved.

The long range test proved that the system was able to work at large distances (188 km) with equal performance.

The optimization results revealed that a small Reed-Solomon protection length is optimal in terms of recovered bytes and overhead.

Raspberry Pis have been used to emulate onboard computers, showing the blazing fast encoding speed of the LPDC-Staircase codec.

The excellent recovery performance of LDPC-Staircase codes has been shown for different code rates and N1 values.

In addition, the minimum received power to ensure a maximum performance as a function of the modulation used has been calculated.

Moreover, it has been determined that the Doppler effect would affect OFDM modulations, but not DSSS modulations.

Finally, the scintillation effect has been studied. In the one hand, it has been shown that the induced phase changes are effectively tracked by the PLLs, introducing negligible errors. On the other hand, the packet loss as a function of the margin above the sensitivity level and S_4 parameter has been determined. The obtained values indicate the optimum coding rate for the LDPC-Staircase codec as a function of the S_4 parameter and link margin.

Chapter 6

Conclusions and Future Work

This chapter contains the conclusions of the thesis and foresees the future work that may be done to improve the current results.

6.1 Conclusions

The goal of this thesis was to implement a high-speed downlink for nanosatellites using WiFi technology. After a careful selection of the off-the-shelf hardware, the appropriate driver and firmware modifications and the implementation of application layer codes a high-speed robust unidirectional link has been achieved.

The tests and benchmarks that have been conducted have shown the feasibility of a long range link implementation, tailored to the computational limits imposed by the cubesat platform.

The use of off-the-shelf WiFi hardware offers a cheap and mature technology capable of working in license-free ISM bands with plenty of bandwidth that support high-rate communications. Open source firmware and drivers gives freedom to temper the default mode of operation and accommodate it to the requirements.

Combining LPDC-Staircase and Reed-Solomon codecs the link becomes robust against long loss-of-signal periods and combats bursts errors while maintaining an encoding complexity and CPU demand perfectly affordable for the spacecraft onboard computers. Studying the scintillation effects and the most common transmission errors the optimal parameters to configure both codes and enhance its performance have been given.

The proper selection of open-source and free of charge codecs is a plus that permits the inspection and enhancement of the software at no cost. The application of LDPC-Staircase and Reed-Solomon codecs is physical layer agnostic and may be potentially implemented over any communication standard. For instance, the LDPC-Staircase codification has been implemented into the ^{3}CAT-2 onboard computer two months before launch, enhancing the link robustness without having to modify other layers.

6.2 Future Work

The LPDC-Staircase + Reed-Solomon combination is ready to be applied to any future system without major modifications. Although the implementation has been extensively tested and no major flaws have been detected, the code should be revised by a senior programmer to be ready for flight.

In order to have a fully unidirectional link the onboard computer should be able to determine the optimal transmission parameters as a function of the satellite-ground station position. That could be done by using the information of the onboard orbit propagator and a predefined table of ground station locations. The system would have to determine for each pass the contact time and elevation above horizon for selecting the appropriate modulation, LDPC-Staircase parameters and transmission time.

Although off-the-shelf WiFi adapters present nice characteristics to be used they are far from optimal implementations. The main problem resides in the physical layer, which is implemented in hardware and is not possible to be modified. WiFi devices use OFDM and DSSS modulations in their physical layer that need relatively high linear amplifiers. Linearity and power efficiency have an inverse relation for power amplifiers meaning that high power amplifiers for WiFi devices are power hungry, having a great impact on the nanosat power budged.

If instead of using off-the-shelf hardware custom transceivers would be made, several aspects could be improved. The Barker/CCK or convolutional codes implemented in the physical layer of the WiFi devices could be replaced by capacity approaching turbocodes which would give a boost in performance. Moreover, the OFDM and DSSS modulations could be changed by constant-envelope modulations that may be driven by high efficiency power amplifiers. The combination of improved channel coding and modulation would positive impact the overall system performance.

A paper explaining the combination of LPDC-Staircase and Reed-Solomon codes for obtaining a robust downlink using COTS WiFi devices has been submitted and is pending of approval and peer review.

Bibliography

[1] *Sputnik launched*. [Online]. Available: http://www.history.com/this-day-in-history/sputnik-launched.

[2] H. Heidt, J. Puig-Suari, A. Moore, S. Nakasuka, and R. Twiggs, "Cubesat: A new generation of picosatellite for education and industry low-cost space experimentation", 2000.

[3] *Cubesat database - swartwout*. [Online]. Available: https://sites.google.com/a/slu.edu/swartwout/home/cubesat-database.

[4] F. Santoni, F. Piergentili, S. Donati, M. Perelli, A. Negri, and M. Marino, "An innovative deployable solar panel system for cubesats", *Acta Astronautica*, vol. 95, pp. 210–217, 2014.

[5] *Deployable solar cells*. [Online]. Available: http://mstl.atl.calpoly.edu/~bklofas/presentations/developersworkshop2012/clark_solar_panels.pdf.

[6] Erik, *Nanosatellites database*, 2016. [Online]. Available: http://nanosats.eu/.

[7] B. Klofas, J. Anderson, and K. Leveque, "A survey of cubesat communication systems", in *5th Annual CubeSat Developers' Workshop*, 2008.

[8] *Be-1/be-2 high speed downlink*. [Online]. Available: http://mstl.atl.calpoly.edu/~bklofas/nsf_comm/20130813_smallsat/astrodev_s-band_brown.pdf.

[9] *Sri high speed downlink*. [Online]. Available: http://mstl.atl.calpoly.edu/~bklofas/nsf_comm/20130813_smallsat/sri_c-band_leveque.pdf.

[10] *Indigo high speed downlink*. [Online]. Available: http://mstl.atl.calpoly.edu/~bklofas/nsf_comm/20130813_smallsat/indigo_cutler.pdf.

[11] *Canopus high speed downlink*. [Online]. Available: http://mstl.atl.calpoly.edu/~bklofas/nsf_comm/20130813_smallsat/canopus_ka-band_king.pdf.

[12] *Syrilinks high speed downlink*. [Online]. Available: http://mstl.atl.calpoly.edu/~bklofas/nsf_comm/20130813_smallsat/syrlinks_x-band_guillois.pdf.

[13] *List of wi-fi chipset vendors*. [Online]. Available: https://wikidevi.com/wiki/list_of_wi-fi_chipset_vendors.

[14] T. Logsdon, *Orbital Mechanics: Theory and Applications*, ser. A Wiley interscience publication. Wiley, 1998, ISBN: 9780471146360.

[15] *Leo sun-sync*. [Online]. Available: http://earthobservatory.nasa.gov/features/orbitscatalog/page2.php.

[16] *Un resolution space debris*. [Online]. Available: http://www.unoosa.org/pdf/gares/ares_62_217e.pdf.

[17] *Nanosatlab*. [Online]. Available: http://www.tsc.upc.edu/nanosatlab/projects.

[18] S. Cakaj, W. Keim, and K. Malaric, "Communications duration with low earth orbiting satellites", in *Proceedings of the 4th IASTED International Conference on Antennas, Radar and Wave Propagation*, Citeseer, 2007.

[19] *Innovative solutions in space*. [Online]. Available: http://www.isispace.nl/.

[20] *Ground station kit*. [Online]. Available: http://www.isispace.nl/product/full-ground-station-kit-s-band/.

[21] *Svalbard*. [Online]. Available: https://en.wikipedia.org/wiki/svalbard.

[22] J. R. Carson, "Reciprocal theorems in radio communication", *Radio Engineers, Proceedings of the Institute of*, vol. 17, no. 6, pp. 952–956, 1929.

[23] *Celestrak*. [Online]. Available: http://celestrak.com/.

[24] K. Riesing, "Orbit determination from two line element sets of iss-deployed cubesats", 2015.

[25] *Atmospheric attenuation*. [Online]. Available: http://www.propagation.gatech.edu/ece6390/project/fall2012/team09/team9geosattech_website_final/satcomwebsite/atmosphericattenuation.html.

[26] M. Kelly, *The Earth's Ionosphere: Plasma Physics and Electrodynamics*, ser. International geophysics series. Elsevier Science, 2012, ISBN: 9780323148054.

[27] *Gnss and ionospheric scintillation*. [Online]. Available: http://www.insidegnss.com/auto/julyaug09-kintner.pdf.

[28] E. Kaplan and C. Hegarty, *Understanding GPS: Principles and Applications*. Artech House, 2005, ISBN: 9781580538954.

[29] M. Aquino, T. Moore, A. Dodson, S. Waugh, J. Souter, and F. S. Rodrigues, "Implications of ionospheric scintillation for gnss users in northern europe", *Journal of Navigation*, vol. 58, no. 02, pp. 241–256, 2005.

[30] R. S. Conker, M. B. El-Arini, C. J. Hegarty, and T. Hsiao, "Modeling the effects of ionospheric scintillation on gps/satellite-based augmentation system availability", *Radio Science*, vol. 38, no. 1, 2003.

[31] T. E. Humphreys, M. L. Psiaki, J. C. Hinks, B. O. Hanlon, and P. M. Kintner Jr, "Simulating ionosphere-induced scintillation for testing gps receiver phase tracking loops", *Selected Topics in Signal Processing, IEEE Journal of*, vol. 3, no. 4, pp. 707–715, 2009.

[32] ——, "Simulating ionosphere-induced scintillation for testing gps receiver phase tracking loops", *Selected Topics in Signal Processing, IEEE Journal of*, vol. 3, no. 4, pp. 707–715, 2009.

[33] M. Reid, *Low-Noise Systems in the Deep Space Network*, ser. JPL Deep-Space Communications and Navigation Series. Wiley, 2008, ISBN: 9780470402283.

[34] S. Gao, K Clark, M. Unwin, J Zackrisson, W. Shiroma, J. Akagi, K Maynard, P Garner, L. Boccia, G. Amendola, *et al.*, "Antennas for modern small satellites", *Antennas and Propagation Magazine, IEEE*, vol. 51, no. 4, pp. 40–56, 2009.

[35] A. Nascetti, E. Pittella, P. Teofilatto, and S. Pisa, "High-gain s-band patch antenna system for earth-observation cubesat satellites", *Antennas and Wireless Propagation Letters, IEEE*, vol. 14, pp. 434–437, 2015.

[36] C. MacGillivray, "Miniature deployable high gain antenna for cubesats", in *2011 CubeSat Developers Workshop April*, 2011, pp. 20–22.

[37] P. Muri, O. Challa, and J. McNair, "Enhancing small satellite communication through effective antenna system design", in *MILITARY COMMUNICATIONS CONFERENCE, 2010-MILCOM 2010*, IEEE, 2010, pp. 347–352.

[38] W. Lemstra, V. Hayes, and J. Groenewegen, *The Innovation Journey of Wi-Fi: The Road to Global Success*. Cambridge University Press, 2010, ISBN: 9781139492577.

[39] *15 years of wi-fi*. [Online]. Available: `http://www.wi-fi.org/discover-wi-fi/15-years-of-wi-fi`.

[40] *Wi-fi: Overview of the 802.11 physical layer and transmitter measurements*. [Online]. Available: `http://www.cnrood.com/public/docs/wifi_physical_layer_and_transm_meas.pdf`.

[41] K. Wesolowski, *Introduction to Digital Communication Systems*. Wiley, 2009, ISBN: 9780470695197.

[42] *Wlan physical layer from teknillinen korkeakoulu university*. [Online]. Available: `http://www.comlab.hut.fi/studies/3240/luentokalvot/5_wlan3.pdf`.

[43] M Sellars and D. Kostas, "Comparison of qpsk/qam ofdm and spread spectrum for the 2-11 ghz pmp bwas", *Response to the Call For Contributions, by IEEE*, vol. 802, pp. 3–00, 2000.

[44] *802.11 phy layers*. [Online]. Available: `http://media.techtarget.com/searchmobilecomputing/downloads/cwap_ch8.pdf`.

[45] *802.11n signal structure*. [Online]. Available: `http://rfmw.em.keysight.com/wireless/helpfiles/n7617a/mimo_ofdm_signal_structure.htm`.

[46] S. Nedevschi and B. University of California, *Maximizing Performance in Long Distance Wireless Networks for Developing Regions*. University of California, Berkeley, 2008, ISBN: 9781109096101.

[47] P. Bhagwat, B. Raman, and D. Sanghi, "Turning 802.11 inside-out", *ACM SIGCOMM Computer Communication Review*, vol. 34, no. 1, pp. 33–38, 2004.

[48] K. Chebrolu, B. Raman, and S. Sen, "Long-distance 802.11 b links: Performance measurements and experience", in *Proceedings of the 12th annual international conference on Mobile computing and networking*, ACM, 2006, pp. 74–85.

[49] C. Oliveira, J. B. Kim, and T. Suda, "Long-range dependence in ieee 802.11 b wireless lan traffic: An empirical study", in *Computer Communications, 2003. CCW 2003. Proceedings. 2003 IEEE 18th Annual Workshop on*, IEEE, 2003, pp. 17–23.

[50] B. A. Chambers, "The grid roofnet: A rooftop ad hoc wireless network", PhD thesis, Massachusetts Institute of Technology, 2002.

[51] R. K. Patra, S. Nedevschi, S. Surana, A. Sheth, L. Subramanian, and E. A. Brewer, "Wildnet: Design and implementation of high performance wifi based long distance networks.", in *NSDI*, vol. 1, 2007, p. 1.

[52] *Wifibroadcast – analog-like transmission of live video data*, 2015. [Online]. Available: https://befinitiv.wordpress.com/wifibroadcast-analog-like-transmission-of-live-video-data/.

[53] R. Shimmin, C. Priscal, K. Oyadomari, M. Murbach, R. Alena, T. Stone, and R. Gilstrap, "The successful phonesat wifi experiment on the soarex-8 flight", *IEEE Aerospace Conference*, 2015.

[54] J. Barceló, B. Bellalta, A. Sfairopoulou, C. Cano, and M. Oliver, "No ack in ieee 802.11 e single-hop ad-hoc voip networks", in *Advances in Ad Hoc Networking*, Springer, 2008, pp. 157–166.

[55] *Linux wifi support*. [Online]. Available: http://linux-wless.passys.nl/.

[56] *Alfa nha datasheet*. [Online]. Available: http://www.giant.in.th/document/awus036nha.pdf.

[57] *Tl-wn722 datasheet*. [Online]. Available: http://www.tp-link.com/resources/document/TL-WN722N_V1_User_Guide_1910010563.pdf.

[58] *Wifistation ext datasheet*. [Online]. Available: http://dl.ubnt.com/wifistation_datasheet.pdf.

[59] *Ar9271 datasheet*. [Online]. Available: https://wikidevi.com/files/atheros/specsheets/ar9271.pdf.

[60] *Skyworks 2576l power amplifier datasheet*. [Online]. Available: http://www.skyworksinc.com/uploads/documents/202428a.pdf.

[61] *Linux wireless networking: A short walk*, 2015. [Online]. Available: https://www.linux.com/blog/linux-wireless-networking-short-walk.

[62] *The madwifi project is history*. [Online]. Available: http://madwifi-project.org/.

[63] *Open ath9k-htc firmware*. [Online]. Available: https://github.com/qca/open-ath9k-htc-firmware.

[64] *Ar7010 datasheet*. [Online]. Available: https://wikidevi.com/files/atheros/specsheets/ar7010ar9287.pdf.

[65] *Libpcap*. [Online]. Available: https://sourceforge.net/projects/libpcap/.

[66] D. J. Barry, *Dealing with the data deluge: Accelerating packet capture*. [Online]. Available: http://www.lovemytool.com/blog/2015/08/dealing-with-the-data-deluge-accelerating-packet-capture-by-dan-joe-barry.html.

[67] *Radiotap*. [Online]. Available: http://www.radiotap.org/.

[68] *Firmware modification*. [Online]. Available: https://befinitiv.wordpress.com/2015/02/22/finding-the-right-wifi-dongle-and-patching-its-kernel-driver-and-firmware/.

[69] C. E. Shannon, "A mathematical theory of communication", *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 5, no. 1, pp. 3–55, 2001.

[70] M. J. Golay, *Notes on digital coding*, 1949.

[71] R. W. Hamming, "Error detecting and error correcting codes", *Bell System technical journal*, vol. 29, no. 2, pp. 147–160, 1950.

[72] I. S. Reed and G. Solomon, "Polynomial codes over certain finite fields", *Journal of the society for industrial and applied mathematics*, vol. 8, no. 2, pp. 300–304, 1960.

[73] E. R. Berlekamp, *Algebraic Coding Theory: Revised Edition*. World Scientific, 2015.

[74] J. L. Massey, "Shift-register synthesis and bch decoding", *Information Theory, IEEE Transactions on*, vol. 15, no. 1, pp. 122–127, 1969.

[75] R. C. Singleton, "Maximum distance q-nary codes", *Information Theory, IEEE Transactions on*, vol. 10, no. 2, pp. 116–118, 1964.

[76] *Reed–solomon error correction*. [Online]. Available: https://en.wikipedia.org/wiki/reedâĂşsolomon_error_correction.

[77] *Ccsds*. [Online]. Available: http://public.ccsds.org/publications/archive/101x0b6s.pdf.

[78] J. Bloemer, M. Kalfane, R. Karp, M. Karpinski, M. Luby, and D. Zuckerman, "An xor-based erasure-resilient coding scheme", 1995.

[79] J. S. Plank *et al.*, "A tutorial on reed-solomon coding for fault-tolerance in raid-like systems", *Softw., Pract. Exper.*, vol. 27, no. 9, pp. 995–1012, 1997.

[80] J. W. Byers, M. Luby, M. Mitzenmacher, and A. Rege, "A digital fountain approach to reliable distribution of bulk data", *ACM SIGCOMM Computer Communication Review*, vol. 28, no. 4, pp. 56–67, 1998.

[81] M. Luby, "Lt codes", in *Null*, IEEE, 2002, p. 271.

[82] A. Shokrollahi, "Raptor codes", *Information Theory, IEEE Transactions on*, vol. 52, no. 6, pp. 2551–2567, 2006.

[83] *Raptor codes - mit*. [Online]. Available: http://web.mit.edu/6.454/www/www_fall_2003/trace/raptor2.pdf.

[84] *Libraptorq*. [Online]. Available: https://www.fenrirproject.org/luker/libraptorq/wikis/libraptorq.pdf.

[85] *Libraptorq webpage*. [Online]. Available: https://github.com/lucafulchir/libraptorq.

[86] *Openrq*. [Online]. Available: http://openrq-team.github.io/openrq/.

[87] M. Luby, *Information additive code generator and decoder for communication systems*, US Patent 6,307,487, 2001.

[88] *Gallager ldpc codes*. [Online]. Available: http://www.rle.mit.edu/rgallager/documents/ldpc.pdf.

[89] V. Roca, M. Cunche, C. Thienot, J. Detchart, and J. Lacan, "Rs+ ldpc-staircase codes for the erasure channel: Standards, usage and performance", in *9th IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob 2013)*, IEEE, 2013.

[90] *Openfec.org*. [Online]. Available: http://openfec.org/.

[91] *Rfc 6816 - simple low-density parity check (ldpc) staircase forward error correction (fec) scheme for fecframe*. [Online]. Available: https://tools.ietf.org/html/rfc6816.

[92] H. Minsky, *Rscode*. [Online]. Available: http://rscode.sourceforge.net/.

[93] *Crc32*. [Online]. Available: http://csbruce.com/software/crc32.c.

[94] *Berkeley packet filter*. [Online]. Available: https://www.kernel.org/doc/documentation/networking/filter.txt.

[95] *Tcpdump/libpcap public repository*. [Online]. Available: http://www.tcpdump.org/.

[96] *Matplotlib: Python plotting*. [Online]. Available: http://matplotlib.org/.

[97] *Iberpix. ortofotos y cartografía raster*. [Online]. Available: http://www.ign.es/iberpix2/visor/.

[98] *Heywhatsthat wisp*. [Online]. Available: https://wisp.heywhatsthat.com/.

[99] *Distance to the horizon*. [Online]. Available: https://web.archive.org/web/20131717132700/http://mintaka.sdsu.edu/gf/explain/atmos_refr/horizon.html.

[100] *Flash earth*. [Online]. Available: http://www.flashearth.com/.

[101] *Alfa network aga-2424t 2.4ghz grid antenna 24dbi*. [Online]. Available: http://www.ciudadwireless.com/network_aga-2424t_2-4ghz_antenna-p-5529.html.

[102] *Ettus research*. [Online]. Available: https://www.ettus.com/.

[103] V. Roca, "Codes al-fec et protocoles de diffusion robuste de contenus: Composants pour des services performants et passant à l'échelle", PhD thesis, Université de Grenoble, 2014.

[104] P. Husted and W. McFarland, *Method and system for noise floor calibration and receive signal strength detection*, US Patent 7,245,893, 2007. [Online]. Available: http://www.google.com/patents/US7245893.

[105] *Minicircuit attenuators*. [Online]. Available: `http://194.75.38.69/products/attenuators_coax_fixed.shtml#fw`.

[106] *Coaxial cable cbl-2m-smsm*. [Online]. Available: `http://194.75.38.69/pdfs/cbl-2m-smsm.pdf`.