Acquisition Techniques in Galileo AltBOC Signals

João Paulo Mateus Pires

Thesis to obtain the Master of Science Degree in

Aerospace Engineering

Supervisor(s): Prof. Fernando Duarte Nunes

Examination Committee
Chairperson: Prof. João Manuel Lage de Miranda Lemos
Supervisor: Prof. Fernando Duarte Nunes
Member of the Committee: Prof. Jose Eduardo Charters Ribeiro da Cunha Sanguino

October 2016
Acknowledgments

First of all, I would like to thank everyone that made possible the conclusion of my master’s degree. A special acknowledgment must be made to Professor Fernando Nunes, without whom the realization of this work would not be possible, thanks to his knowledge, availability, contribution and guidance.

I also would like to express my gratitude to my family, who gave me all the support imaginable during not only the realization of this work but also during all my academic years.
Resumo

O objectivo do presente trabalho é apresentar um estudo completo da modulação AltBOC(15,10), a ser usada na banda E5 do sistema de navegação por satélite Galileo, e descrever algoritmos de aquisição adequados à realização da aquisição dos sinais do sistema Galileo presentes na banda E5.

Esta tese começa por apresentar a descrição completa da modulação AltBOC. Logo após, são apresentados o hardware indispensável à realização da aquisição por software e os princípios básicos da aquisição.

De seguida, são apresentados os métodos de aquisição mais adequados à modulação. É realizada a análise destes mesmos algoritmos no que toca ao tempo de aquisição e à probabilidade de correta deteção para diferentes valores de $C/N_0$, usando dados simulados, que se aproximam o mais possível de dados reais. Todos os algoritmos são analisados em aquisições do tipo SSB, DSB e direta.

Palavras-chave: GNSS, Aquisição, AltBOC, Galileo, banda E5.
Abstract

The objective of this work is to present a complete study of the AltBOC(15,10) modulation and to describe acquisition algorithms suitable to perform a software acquisition on the Galileo E5 band signals.

The thesis starts by the full description of the AltBOC modulation. Then, the minimum hardware necessary to perform the acquisition and the basic principles of acquisition is presented.

Next, the most suitable acquisition algorithms for the AltBOC modulation are presented and described. These algorithms are then analyzed in terms of acquisition time and the probability of correct detection for different C/N0, using simulated data as close as possible to the real data. All the algorithms are analyzed in conditions of Single Side Band, Double Side Band and Direct acquisition.

Keywords: GNSS, Acquisition, AltBOC, Galileo, E5 band.
# Contents

Contents

Acknowledgments ................................................................. iii
Resumo ........................................................................ v
Abstract ........................................................................ vii
List of Tables ..................................................................... xiii
List of Figures .................................................................... xv
Glossary ........................................................................... xvii

1 Introduction ...................................................................... 1
  1.1 Global Navigation Satellite Systems ............................. 1
  1.2 Basic Operation Concept ............................................. 2
  1.3 Galileo System Overview ............................................. 3
    1.3.1 Frequency Plans and Services ................................. 4
  1.4 Motivation ................................................................... 5
  1.5 Thesis Objectives ......................................................... 6
  1.6 Thesis Outline ............................................................. 6

2 Galileo E5 AltBOC Signal .................................................. 7
  2.1 E5 Signal Overview ...................................................... 7
    2.1.1 Multiple Access Schemes ........................................... 8
    2.1.2 PRN Code Generation ........................................... 9
    2.1.3 Sub-carrier Generation .......................................... 12
    2.1.4 Navigation Data Generation ................................. 13
  2.2 AltBOC Modulation ....................................................... 13
    2.2.1 BOC modulation .................................................. 14
    2.2.2 Standard AltBOC modulation ................................. 15
    2.2.3 E5 AltBOC(15,10) ................................................ 18
  2.3 Receiver Hardware Considerations ................................. 23
    2.3.1 Single Band Receiver ........................................... 23
    2.3.2 Separate Double Band Receiver ............................. 24
    2.3.3 Coherent Double Band receiver ............................. 25
    2.3.4 Hardware Parameters .......................................... 25
## List of Tables

1.1 Galileo navigation signal parameters .................................................. 5

2.1 E5 Chip Rates and Symbol Rates ......................................................... 8

2.2 E5 Code Lengths. ................................................................. 9

2.3 E5 Primary Codes Specifications. ....................................................... 10

2.4 Logic to Signal Level Assignment. ..................................................... 12

2.5 AltBOC Sub-carrier Coefficients. ..................................................... 13

2.6 Look-up Table for AltBOC Phase States. ............................................ 23

5.1 Parameters used in the classic acquisition. ........................................... 57

5.2 Time results for the classic acquisition. .............................................. 59

5.3 Parameters used in the parallel search in frequency domain. .................. 63

5.4 Time results for the parallel search in frequency domain acquisition. ....... 63

5.5 Parameters used in the parallel search in time domain. ......................... 66

5.6 Time results for the parallel search in time domain acquisition. .............. 66

5.7 Parameters used in the DBZPTI algorithm. ......................................... 68

5.8 Time results for the DBZPTI acquisition. ......................................... 69

5.9 Normalized MAT for all the studied algorithms. .................................... 71
List of Figures

1.1 GNSS triangulation using three satellites. .................................................. 2
1.2 Galileo System Segments. .............................................................................. 3
1.3 Galileo Frequency Plan. ................................................................................ 4

2.1 Spectral scheme for Galileo E5 band. ............................................................... 7
2.2 Multiple access schemes. ................................................................................ 8
2.3 Tiered Codes Generation. ................................................................................ 9
2.4 LFSR Based Code Generator for Truncated and Combined M-sequences. .... 10
2.5 Correlation properties of the primary codes used in the E5 band. ............... 11
2.6 Auto-correlation property of a secondary code used in the E5 band. .......... 11
2.7 One Period of the Two Sub-carrier Functions Involved in AltBOC Modulation. 13
2.8 Subcarrier function for BOC modulation. ....................................................... 14
2.9 PSD of $s(t)$, before and after the BOC Modulation. ..................................... 15
2.10 Spectrum of the complex exponential $e^{r(t)}$. ............................................. 16
2.11 Spectrum of the CBOC Modulated Signal. .................................................... 17
2.12 Scattered plot for the standard AltBOC modulation. .................................... 18
2.13 Modulation Scheme for the E5 Signal. ......................................................... 18
2.14 Spectrum of one the complex exponentials used in E5 AltBOC modulation. .... 20
2.15 PSD of the AltBOC(15,10) signal. ................................................................. 20
2.16 Normalized ACF of the AltBOC(15,10) signal. ............................................. 21
2.17 Normalized ACF of the BPSK signal. ............................................................ 21
2.18 8-PSK Phase-State Diagram of E5 AltBOC Signal. ....................................... 22
2.19 Architecture of a Single Band Receiver Appropriate for the Acquisition on the E5a Sideband. ............................................................ 24
2.20 Architecture of a Separate Double Band Receiver. ....................................... 24
2.21 Architecture of a Coherent Double Band Receiver. ...................................... 25
2.22 Normalized ACF of the BPSK modulation. .................................................... 26
2.23 Normalized ACF of the AltBOC(15,10) modulation. .................................... 27

3.1 Carrier Wipe-off. ......................................................................................... 30
3.2 Code Wipe-off. ........................................................................................... 31
3.3 Output of the Envelope Detector for $\hat{\delta} = \delta_d$. ....................................... 32
3.4 Output of the Envelope Detector for $\hat{\tau} = \tau$. 33
3.5 Representation of the Search Space. 33

4.1 Architecture of the Serial Acquisition Algorithm for a Single Channel. 42
4.2 Architecture of the Serial Acquisition Algorithm for both Channels of a Sideband. 43
4.3 Architecture of the Parallel Acquisition in Frequency Domain Algorithm. 43
4.4 Architecture of the Parallel Acquisition in Time Domain Algorithm. 45
4.5 Pre-processing of the incoming signal. 46
4.6 Local replica code. 46
4.7 Partial circular correlation. 46
4.8 Circular correlation for all delays. 46
4.9 Zero padding and FFT application. 47
4.10 CL of the Galileo E5 Secondary Codes with length 100. 49
4.11 Architecture of the Proposed Algorithm. 49
4.12 Secondary Code Evolutionary Tree at the Third Iteration. 50
4.13 Architecture of the SPC Algorithm. 52
4.14 Normalized Correlation Values for the first E5a-Q Code. 52

5.1 SSB classical acquisition results. 59
5.2 SSB classical acquisition results after a 10 non-coherent integrations (K=10). 60
5.3 Histogram of the SSB classical acquisition results at $C/N_0 = 46$dB.Hz. 60
5.4 Histograms of the classical acquisition results at $C/N_0 = 46$dB.Hz. 61
5.5 Sensitivity of the Classical Acquisition. 61
5.6 Example of a direct acquisition performed with a progressive acquisition technique. 62
5.7 Zoomed Normalized Correlation of the Direct Acquisition Peak. 62
5.8 SSB Algorithm Output. 64
5.9 Histograms of the parallel search in frequency domain acquisition results at $C/N_0 = 46$dB.Hz. 65
5.10 Sensitivity of the Parallel Search in Frequency Domain Algorithm. 65
5.11 SSB Algorithm Output. 67
5.12 Histograms of the parallel search in time domain acquisition results at $C/N_0 = 46$ dB.Hz. 67
5.13 Sensitivity of the Parallel Search in Time Domain Algorithm. 68
5.14 SSB Algorithm Output. 69
5.15 Histograms of the DPZPTI algorithm at $C/N_0 = 46$ dB.Hz. 70
5.16 Sensitivity of the DBZPTI Algorithm. 70
5.17 Sensitivity comparison of the different algorithms. 71
5.18 Probability of detection versus $C/N_0$ using one channel acquisition. 72
5.19 Acquisition time comparison between coherent and non-coherent combining. 73
# Glossary

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACF</td>
<td>Auto Correlation Function</td>
</tr>
<tr>
<td>ADC</td>
<td>Analog-to-Digital Converter</td>
</tr>
<tr>
<td>ARNS</td>
<td>Aeronautical Radio Navigation Services</td>
</tr>
<tr>
<td>AWGN</td>
<td>Additive White Gaussian Noise</td>
</tr>
<tr>
<td>AltBOC</td>
<td>Alternative Binary Offset Carrier</td>
</tr>
<tr>
<td>BOC</td>
<td>Binary Offset Carrier</td>
</tr>
<tr>
<td>BPSK</td>
<td>Binary Phase Shift-Keying</td>
</tr>
<tr>
<td>C/N&lt;sub&gt;0&lt;/sub&gt;</td>
<td>Carrier-to-Noise-density ratio</td>
</tr>
<tr>
<td>CBOC</td>
<td>Complex Binary Offset Carrier</td>
</tr>
<tr>
<td>CDMA</td>
<td>Code Division Multiple Access</td>
</tr>
<tr>
<td>CL</td>
<td>Characteristic Length</td>
</tr>
<tr>
<td>CS</td>
<td>Commercial Service</td>
</tr>
<tr>
<td>DBZPTI</td>
<td>Double Block Zero Padding Transition Insensitive</td>
</tr>
<tr>
<td>DBZP</td>
<td>Double Block Zero Padding</td>
</tr>
<tr>
<td>DLL</td>
<td>Delay Locked Loop</td>
</tr>
<tr>
<td>DSB</td>
<td>Double Side Band</td>
</tr>
<tr>
<td>FDMA</td>
<td>Frequency Division Multiple Access</td>
</tr>
<tr>
<td>FFT</td>
<td>Fast Fourier Transformation</td>
</tr>
<tr>
<td>FIC</td>
<td>Full-band Independent Code</td>
</tr>
<tr>
<td>FLL</td>
<td>Frequency Locked Loop</td>
</tr>
<tr>
<td>FOC</td>
<td>Full Operational Capability</td>
</tr>
<tr>
<td>GLONASS</td>
<td>GLObal NAvigation Satellite System</td>
</tr>
<tr>
<td>GNSS</td>
<td>Global Navigation Satellite System</td>
</tr>
<tr>
<td>GPS</td>
<td>Global Positioning System</td>
</tr>
<tr>
<td>IFFT</td>
<td>Inverse Fast Fourier Transform</td>
</tr>
<tr>
<td>IF</td>
<td>Intermediate Frequency</td>
</tr>
<tr>
<td>IOV</td>
<td>In-Orbit Validation</td>
</tr>
<tr>
<td>LFSR</td>
<td>Linear Feedback Shift Register</td>
</tr>
<tr>
<td>LUT</td>
<td>Look-UP Table</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Full Form</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------</td>
</tr>
<tr>
<td>MAT</td>
<td>Mean Acquisition Time</td>
</tr>
<tr>
<td>MEO</td>
<td>Medium Earth Orbit</td>
</tr>
<tr>
<td>NBP</td>
<td>Narrow-Band Power</td>
</tr>
<tr>
<td>OS</td>
<td>Open Service</td>
</tr>
<tr>
<td>PDF</td>
<td>Probability Density Function</td>
</tr>
<tr>
<td>PLL</td>
<td>Phase Locked Loop</td>
</tr>
<tr>
<td>PRN</td>
<td>Pseudo Random Noise</td>
</tr>
<tr>
<td>PRS</td>
<td>Public Regulation Service</td>
</tr>
<tr>
<td>PSD</td>
<td>Power Spectral Density</td>
</tr>
<tr>
<td>RF</td>
<td>Radio frequency</td>
</tr>
<tr>
<td>RHCP</td>
<td>Right-Hand Circular Polarized</td>
</tr>
<tr>
<td>SARS</td>
<td>Search and Rescue Service</td>
</tr>
<tr>
<td>SNR</td>
<td>Signal-to-Noise Ratio</td>
</tr>
<tr>
<td>SPC</td>
<td>Sub-carrier Phase Cancellation</td>
</tr>
<tr>
<td>SSB</td>
<td>Single Side Band</td>
</tr>
<tr>
<td>SVID</td>
<td>Space Vehicle ID</td>
</tr>
<tr>
<td>SoL</td>
<td>Safety-of-Life</td>
</tr>
<tr>
<td>TDMA</td>
<td>Time Division Multiple Access</td>
</tr>
<tr>
<td>WBP</td>
<td>Wide-Band Power</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

In this first chapter of the work, a brief description of the Global Satellite Navigation Systems, in particular of the developing Galileo system, is presented. In addition, the thesis' motivation, objectives and outline are explained in sections 1.4, 1.5 and 1.6, respectively.

1.1 Global Navigation Satellite Systems

A Global Navigation Satellite System (GNSS) is a navigation system that has global coverage and allows small receivers to determine their exact location using line-of-sight satellite radio transmitted signals. These signals can also be used for time synchronization, since they also transmit time information with high precision.

The first idea to deploy a navigation system using satellite signals came from the United States department of defense during the decade of 1960. They needed an always available and capable of high precision navigation system, so they developed the Transit system [1].

The Transit was composed by seven satellites in low-altitude polar orbits. This system was based on the Doppler shift effect: the satellites traveled in known paths and transmitted signals with known frequency. Due to the movement of the satellite with respect to the receiver, the received signal would have different frequency from the one transmitted. By monitoring this Doppler frequency shift, and knowing all the satellite positions at a given time, it was possible to determine the particular position of the receiver [2].

In the following years, research continued to be made to improve this kind of system, always with a military purpose in sight and the American Global Positioning System (GPS) and the Russian GLObal NAVigation Satellite System (GLONASS) were fully operational in 1995. Soon people realized the enormous amount of possible applications of these systems, so they began to be available to the civil population, even though the control always remained and still is in possession of the military.

As of today, these two systems (GPS and GLONASS) are the only fully operational GNSS. However, the European Galileo is being deployed and the chinese Beidou is being expanded from a local system to a global system. Both are expected to be fully functional in 2020 [2].
1.2 Basic Operation Concept

The modern GNSS operate in a different way from the Doppler shift ones. They are based on the \textit{time of arrival} of the signals transmitted by the satellites. Satellites transmit \textit{pseudo-random noise} (PRN) codes and the receiver calculates its position by evaluating the time the signal takes to travel from the satellite to the receiver. This is possible because the satellites send the signal time of transmission with great precision, thanks to great accuracy atomic clocks on-board the satellites. The satellites also transmit navigation data, which gives information relative to the orbits of the satellites, and can be used to determine the position of the satellite at the signal time of transmission [3].

The receiver uses the difference between the time of arrival and time of transmission to determine the distance from its position to a given satellite. Since the position of each satellite is well known, each measurement places the receiver in a spherical shell at the measured distance. By combining the measures from a variety of satellites, the receiver can estimate its position using triangulation techniques. Combining the measures from three satellites, the receiver can estimate to be located in two different locations (the intersection between three spherical shells, results in two distinct points). If the receiver has information that it is on the surface of the Earth (or at least close to it), it can exclude one of those locations, which leaves only one possible estimate. The triangulation using the signal from threes satellites is exemplified in Figure 1.1.

![Triangulation](image)

\textit{Figure 1.1: GNSS triangulation using three satellites [4].}

Since the receiver's clock is not synchronized with the atomic clocks of the satellites, it is necessary to receive information from an additional satellite to estimate the difference of times between the clocks. In conclusion, a receiver needs to have at least four satellites in line-of-sight to determine, without ambiguity, its own position.

These measurements are affected from several sources of errors, such as ionospheric and tropospheric delays and multipath effects. These effects can be reduced using complex data processing.
1.3 Galileo System Overview

Galileo is an ongoing project being developed by the European Union, European Space Agency and several other European organizations, whose goal is to launch a European fully functional GNSS under civil control. This system is expected to be fully operational in 2020.

Galileo is Europe’s attempt to achieve a state-of-the-art GNSS that provides a highly accurate and global positioning service under civil control. It will be interoperable with the GPS and GLONASS, the other two current GNSS, while providing autonomous navigation and positioning services. A receiver will be able to combine the use of the Galileo signals with the rest of the GNSS, which will improve the performance of every GNSS receiver in the world [5].

The biggest improvement over the current GNSS is the availability of dual frequencies in standard services, which will make possible to have precision in positioning down to the meter, which is unprecedented in the public available services of the current GNSS.

Similarly to other GNSS, Galileo will be composed by three segments, as represented in Figure 1.2:

- Space segment.
- Ground segment.
- User segment.

![Figure 1.2: Galileo System Segments [6].](image)

The fully deployed space segment will consist of 24 operational satellites and up to 6 active spares, positioned in three circular Medium Earth Orbit (MEO) planes. Each orbit has a nominal average semi-major axis of 29600 km, and an inclination of 56 degrees with reference to the equatorial plane. Once this is complete, this system should provide good coverage up to latitudes of 75 degrees, either North or South [5].

The ground segment is composed by the ground control segment for operations, which can be used to determine orbit position and time and to monitor the integrity\(^1\) of the system.

The information provided by these two first segments will be used by the user segment, consisting of all users on land, on water, in the air and in outer space.

\(^1\)Integrity is the ability to notify that a given satellite is not in conditions to be used.
The Galileo project has two main steps: the In-Orbit Validation (IOV) and the Full Operational Capability (FOC). The first of these steps, consisting of 4 satellites, has been successfully complete, and the second step is already underway. As of the time of this work, there are 14, of the 30 foreseen, satellites in orbit. This last step is supposed to be concluded in 2019 when 24 satellites are planned to be in orbit. The complete 30-satellite Galileo system (24 operational and 6 active spares) is expected by 2020 [7][8].

1.3.1 Frequency Plans and Services

The Galileo navigation signals are transmitted in four frequency bands, as indicated in Figure 1.3. These four bands are called E5a, E5b, E6 and E1 [5].

The frequency bands have been selected in the allocated spectrum for Radio Navigation Satellite Services (RNSS) and in addition to that, the E5a, E5b and E1 bands are included in the allocated spectrum for Aeronautical Radio Navigation Services (ARNS), employed by Civil-Aviation users, and allow dedicated safety-critical applications [5]. In addition, the E5a and the E1 are overlapped with the GPS L5 and L1 bands, respectively.

All satellites share the same frequency bands by employing Code Division Modulation Access (CDMA) techniques. Spread Spectrum signals will be transmitted that include quasi-orthogonal ranging codes, which are different for every signal component, every signal and every satellite[9].

Galileo has 5 types of services planned [10]:

- Open Service (OS): it is the base free service, thought for general civil applications (the same way as the GPS L1 band, but with better precision). No authorization is required but this service doesn’t include integrity checking.
- Safety-of-Life (SoL): it has the same performance as the OS, but includes a global integrity checking.
- Commercial Service (CS): this service will present encrypted signals with higher transmission rate, which achieves better precision. This service will be available for the general public through
payment of a fee.

- Public Regulation Service (PRS): this service is to be used by applications which require a high level of continuity. This service is encrypted and projected to have good anti-jamming mechanisms.

- Search and Rescue Service (SARS): the satellites will be able to receive distress signals from boats, planes and people and forward these signals to emergency stations. The received signals will allow the emergency stations to locate the site of the problem with great accuracy.

Different modulation types will be employed for each of the bands: E5, E6 and E1. In detail, 10 different signals will be transmitted, with 4 of those being pilot signals\(^2\). The main parameters are shown in Table 1.1. Note that the signals used for PRS are encrypted and, therefore, their signal parameters are not all available to the public as the parameters for the other signals are.

<table>
<thead>
<tr>
<th>Signal</th>
<th>Channel</th>
<th>Modulation</th>
<th>Chip Rate [Mcps]</th>
<th>Data Symbol Rate [sps]</th>
<th>Total Received Minimum Power [dBW]</th>
</tr>
</thead>
<tbody>
<tr>
<td>E5</td>
<td>E5a Data</td>
<td>AltBOC(15,10)</td>
<td>10.23</td>
<td>50</td>
<td>-155</td>
</tr>
<tr>
<td></td>
<td>E5a Pilot</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>E5b Data</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>E5b Pilot</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>E6</td>
<td>E6-B Data OS</td>
<td>BPSK(5)</td>
<td>5.115</td>
<td>1000</td>
<td>-155</td>
</tr>
<tr>
<td></td>
<td>E6-C Pilot OS</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>E6-A Data PRS</td>
<td>BOC(_{cos}(10,5))</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>E1</td>
<td>E1-B Data OS</td>
<td>CBOC(6,1,1/11)</td>
<td>1.023</td>
<td>250</td>
<td>-157</td>
</tr>
<tr>
<td></td>
<td>E1-C Pilot OS</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>E1-A Data PRS</td>
<td>BOC(_{cos}(15,2.5))</td>
<td>2.5575</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 1.1: Galileo navigation signal parameters [5] [11].

1.4 Motivation

This thesis is motivated by the ongoing deployment of the Galileo system. In the next years, when the system becomes operational, there will be a need to update the current GNSS receivers, so that they can benefit not only from the current GNSS signals but also from the new signals provided by the Galileo satellites.

In particular, the most promising signal that will be transmitted is the AltBOC modulated one, present in the E5 band. In theory, the AltBOC modulation can lead to formidable precision of positioning compared to the current GNSS signals, even in the presence of multipath errors. Due to the attractiveness of the AltBOC signal, also the Chinese navigation system, Beidou, plans to transmit signals with the exact same modulation [8].

Therefore, the AltBOC signal acquisition is, and will be in the next few years, one of the most interesting problems in the study of GNSS receivers.

\(^2\)Pilot signals do not include navigation data.
1.5 Thesis Objectives

The first objective of this work is the study of the E5 band signal, in particular, the study of the Galileo AltBOC(15,10) modulation. The second objective is to analyze the performance of a variety of methods and algorithms suitable to achieve signal acquisition in a software receiver. In this work, it is shown that a receiver cannot be 100% software, so the minimum hardware needed is also presented.

1.6 Thesis Outline

The present thesis is structured as follows:

- Chapter 1 gives an introduction to the field of study where the subject of this work is inserted. In addition, the Galileo system is briefly presented.

- Chapter 2 presents a deep description of the signals (and their components) transmitted in the E5 band. The study mainly focuses on the AltBOC modulation and it is shown how the modulation was derived from the conventional BOC modulation. At the end of the chapter, the minimum hardware necessary to perform a software acquisition is presented, as well as three possible hardware configurations.

- Chapter 3 gives an introduction to GNSS signal acquisition. The concepts of correlation, search space, decision threshold and combining are introduced.

- Chapter 4 classifies the main methods for acquisition by the components used and by how the search is made in the acquisition grid. Furthermore, the most suitable methods for software acquisition are presented and discussed.

- Chapter 5 analyzes the performance of the algorithms introduced in chapter 4. This analysis consists of the analysis of the mean acquisition time (MAT) and the computation of probabilities of detection for various carrier-to-noise ratios. Furthermore, a study is made on the effect of the integration time.

- Chapter 6 summarizes the results obtained in this work and concludes on the results.
Chapter 2

Galileo E5 AltBOC Signal

The signal transmitted in the E5 band uses the modern \textit{AltBOC} modulation. This modulation scheme is the result of the optimization of the BOC modulation by the \textit{European Space Agency}.

The E5 band is composed by 2 separate sidebands (E5a and E5b), each having two different signals (1 in-phase and 1 in quadrature). This allows each sideband to be processed individually, as a BPSK signal by a simple receiver, or together, which can lead to good performances in terms of noise [9].

In this chapter, it is presented, in detail, the full description of the signal components in the E5 band present in the Galileo AltBOC modulation. The generation of the ranging codes, sub-carriers and navigation data is presented in sections, 2.1.2, 2.1.3 and 2.1.4. In section 2.2.1 the BOC modulation is introduced and, in the next sections, it is derived the AltBOC modulation used in the E5 band which is described in section 2.2.3. At the end of the chapter (section 2.3), the minimum hardware necessary to perform a software acquisition is presented, as well as three possible hardware configurations.

2.1 E5 Signal Overview

The data transmitted by the satellites in the E5 band uses the AltBOC modulation. The modulated signal is transmitted around a carrier frequency \((f_{E5})\) of 1191.795 MHz over a bandwidth of 51.150 MHz in a Right-Hand Circular Polarization (RHCP) [5].

The E5 band can be divided in two sidebands, as shown in Figure 2.1:

- E5a band: central frequency \((f_{E5a})\) of 1176.45 MHz.
- E5b band: central frequency \((f_{E5b})\) of 1207.14 MHz.

![Figure 2.1: Spectral scheme for Galileo E5 band [9].](image)
In total, four signals are transmitted in the E5 band: two in the E5a sideband and two in the E5b sideband. In each sideband there is an in-phase and a quadrature signal. The in-phase components carry navigation messages: the F/NAV for the Open Service (E5a-I) and the I/NAV for the Safety-of-Life Service (E5b-I). The quadrature components are pilot signals that don’t carry any navigation data, which are mainly important for the acquisition and tracking procedures. The features of these signal components are provided in Table 2.1.

<table>
<thead>
<tr>
<th>Signal</th>
<th>Component</th>
<th>Code Chip-Rate [Mchip/s]</th>
<th>Symbol-Rate [sps]</th>
<th>Data Service</th>
</tr>
</thead>
<tbody>
<tr>
<td>E5a</td>
<td>I</td>
<td>10.230</td>
<td>50</td>
<td>F/NAV</td>
</tr>
<tr>
<td></td>
<td>Q</td>
<td>10.230</td>
<td>No data</td>
<td>-</td>
</tr>
<tr>
<td>E5b</td>
<td>I</td>
<td>10.230</td>
<td>250</td>
<td>I/NAV</td>
</tr>
<tr>
<td></td>
<td>Q</td>
<td>10.230</td>
<td>No data</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 2.1: E5 Chip Rates and Symbol Rates [5]

2.1.1 Multiple Access Schemes

Signals generated by multiple sources can share the same channels if they are orthogonal. There are 3 main methods to make the signals orthogonal, as shown in Figure 2.2 [10].

![Multiple access schemes](image)

The first method is the Frequency Division Multiple Access (FDMA). In this, a small part of the spectrum is allocated to a given source during the whole transmission.

The second one is called Time Division Multiple Access (TDMA). In this method, each source uses all spectrum during a small period of time (time slot), which is different for every source. Here, the signals are not transmitted simultaneously but “one at a time”. The basis of this method are non-overlapping rectangular pulses.

The last, and the one used in most GNSS, including the Galileo system, is the Code Division Multiple Access (CDMA). This scheme makes 2 signals orthogonal by multiplying them with different orthogonal spreading codes. This allows all users to simultaneously use the full spectrum at all times. Although, it is important to note that, contrarily to the TMDA and FDMA methods, this scheme demands a minimum
coordination (in time or frequency) between the multiple users [10].

### 2.1.2 PRN Code Generation

Each one of the signals is modulated with a different PRN code that acts as a spreading code in a CDMA system. These codes, currently defined in [5], follow a tiered code structure. In this structure, each signal has individual primary and secondary codes and the code lengths are presented in Table 2.2. Every code has the same chip rate of \( R_C = 10.23 \text{ Mchip/s} \).

<table>
<thead>
<tr>
<th>Signal Component</th>
<th>Tiered Code Period [ms]</th>
<th>Code Length (chips)</th>
</tr>
</thead>
<tbody>
<tr>
<td>E5a-I</td>
<td>20</td>
<td>10230 20</td>
</tr>
<tr>
<td>E5a-Q</td>
<td>100</td>
<td>10230 100</td>
</tr>
<tr>
<td>E5b-I</td>
<td>4</td>
<td>10230 4</td>
</tr>
<tr>
<td>E5b-Q</td>
<td>100</td>
<td>10230 100</td>
</tr>
</tbody>
</table>

Table 2.2: E5 Code Lengths [5].

The tiered code construction allows the generation of long spreading codes from shorter primary and secondary codes. The secondary code is used to modify successive repetitions of the primary code, as shown in Figure 2.3, for a primary code of length \( N \) and chip rate \( f_c \), and a secondary code of length \( N_S \) and chip rate \( f_{cs} = f_c / N \). In this construction, one chip of the secondary code is synchronized and has the same period as a repetition of the entire primary code, that corresponds to 1ms.

![Figure 2.3: Tiered Codes Generation](image)

**Primary Code**

The primary codes can be generated in two different ways: they can be stored in memory or can be generated by linear feedback shift registers (LFSR) [5].

The first approach is pretty straightforward: the codes are stored in memory and then read whenever they are needed. Instead, the method used in the present work relies on the usage of LFSR. The codes used in Galileo can be generated as combinations of two M-sequences, being the shift register lengths, \( M \), equal to 14.

Figure 2.4 shows the standard implementation of the LFSR used to generate the truncated and combined M sequences used in the E5 Galileo signals. Two parallel shift registers are used: base register 1 and base register 2. The primary code output sequence is the exclusive OR of base register 1 and 2 output sequences, the shift between these two sequences is zero. Each shift register \( i \) (\( i = 1 \) for base register 1 and \( i = 2 \) base register 2) of length \( R \) is fed back with a particular set of feedback taps \( (\alpha_{i,j})_{j=1...R} = \).
and its content is represented by a vector \( (c_{i,j})_{j=1,...,R} = [c_{i,1}, c_{i,2}, ..., c_{i,R}] \). For truncation to primary code length \( N \), the content of the two shift registers is reinitialized (reset) after \( N \) cycles with the so-called start-values \( (s_{i,j})_{j=1,...,R} = [s_{i,1}, s_{i,2}, ..., s_{i,R}] \) [5].

![LFSR Based Code Generator for Truncated and Combined M-sequences](image)

Figure 2.4: LFSR Based Code Generator for Truncated and Combined M-sequences [5].

The start-values of the two shift registers are individual for each one of the satellites and each of the components and those are presented in [5]. The feedback taps are the same for every satellite but they differ from component to component. Table 2.3 presents the E5 primary code specifications.

<table>
<thead>
<tr>
<th>Signal Component</th>
<th>Shift Register Length (polynomial order)</th>
<th>Feedback Taps (octal)</th>
<th>Register 1</th>
<th>Register 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>E5a-I</td>
<td>14</td>
<td>40503</td>
<td>50661</td>
<td></td>
</tr>
<tr>
<td>E5a-Q</td>
<td>14</td>
<td>40503</td>
<td>50661</td>
<td></td>
</tr>
<tr>
<td>E5b-I</td>
<td>14</td>
<td>64021</td>
<td>51445</td>
<td></td>
</tr>
<tr>
<td>E5b-Q</td>
<td>14</td>
<td>64021</td>
<td>43143</td>
<td></td>
</tr>
</tbody>
</table>

Table 2.3: E5 Primary Codes Specifications [5].

In Figure 2.5, the correlation properties of the primary codes can be analyzed. The auto-correlation and the cross-correlation, normalized to the primary code length, were made for the first E5a-I code and for the first two E5a-I codes, respectively. This was an arbitrary choice, as if other random codes were chosen, the results would have been very similar.

From Figure 2.5, the quasi-orthogonality of the codes can be seen, as there is a strong auto-correlation peak when the code is overlaid with its replica. A result of the codes not being exactly orthogonal is that the cross-correlation is not exactly null, which may induce a false acquisition lock. However, since the cross-correlation values are considerably inferior (not greater than 5\%) to auto-correlation with delay 0, the error they introduce is usually neglected because it is inferior to the ones caused by noise and other error sources.
The normalized auto- and cross-correlations in Figure 2.5 are computed as:

$$C_{XY}(k) = \frac{1}{N} \sum_{n=1}^{N} x(n) \cdot y(n-k),$$

where $x(n) = \pm 1$ and $y(n) = \pm 1$ are chips and $N = 10230$ is the number of chips in a code period.

**Secondary Code**

The secondary codes are fixed sequences of bits that need to be stored in memory, as they cannot be generated by shift registers. As Table 2.2 shows, the length of the secondary codes is not the same for the 4 channel signals. All secondary codes can be found in [5].

In Figure 2.6, the auto-correlation of the first E5a-Q secondary code normalized to its length. Similarly to what was explained for the primary code, the choice of the code analyzed was random, and any other choice of code (from the codes with length 100) would lead to similar results.
From Figure 2.6, it can be seen that the secondary codes also have a strong correlation peak when there is no delay between the code and its replica, and a near null correlation for the rest of the delays. This property is really useful during the secondary code acquisition.

In this section only the secondary codes with length 100 (the ones reserved for the pilot channels) were analyzed since they are the ones that need to be acquired, as they are the longest. Because all codes are synchronized and the other codes have lengths that are dividers of 100 (4 for the E5a-I and 20 for the E5b-I), their code delay can be determined from the code delay of the longer secondary codes.

### Code Assignment

According to [5], the code assignment is made as follows:

- The E5a-I, E5a-Q, E5b-I and E5b-Q primary codes are allocated to the space vehicle ID's (SVID) by assigning the SVID \( n \) the corresponding E5a-I, E5a-Q, E5b-I and E5b-Q primary code number \( n \).
- The secondary codes \( CS_{201} \) for the E5a-I and \( CS_{41} \) for the E5b-I are assigned to all satellites.
- The secondary code \( CS_{100_n} \) for the E5a-Q is assigned to the satellite with SVID \( n \).
- The secondary code \( CS_{100(50+n)} \) for the E5b-Q is assigned to the satellite with SVID \( n \).

The correspondence between the logic level code bits used to modulate the signal and the signal level is according to the values stated in Table 2.4. This makes the obtained signals BPSK modulated. Also the exclusive OR of two signals is readily obtained from the product of signal levels [12].

<table>
<thead>
<tr>
<th>Logic Level</th>
<th>Signal Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-1.0</td>
</tr>
<tr>
<td>0</td>
<td>+1.0</td>
</tr>
</tbody>
</table>

**Table 2.4:** Logic to Signal Level Assignment [5].

### 2.1.3 Sub-carrier Generation

The sub-carriers used in the AltBOC modulation are two four-valued functions, \( sc_{E5-S(t)} \) and \( sc_{E5-P(t)} \) (see Figure 2.7). The S and P indexes denote whether the sub-carrier is used for single (S) signals or product (P) signals (see section 2.2.3) [5]. The \( sc_{E5-S(t)} \) signal is an approximation to the cosine function.

The sub-carriers can be described by the following expressions, where the sub-carrier period, \( T_{S,E5} \), is divided in 8 equal sub-periods indicated by the index \( i \), as:

\[
sc_{E5-S}(t) = \sum_{i=-\infty}^{\infty} AS_{[i]} \cdot \text{rect}_{T_{S,E5}}(t - i.T_{S,E5}/8)
\]

\[
sc_{E5-P}(t) = \sum_{i=-\infty}^{\infty} AP_{[i]} \cdot \text{rect}_{T_{S,E5}}(t - i.T_{S,E5}/8).
\]

(2.2)

The coefficients \( AS_i \) and \( AP_i \) are presented in Table 2.5.

---

**Table 2.5:** Coefficients for Sub-carrier Generation.
### Table 2.5: AltBOC Sub-carrier Coefficients [5].

<table>
<thead>
<tr>
<th>i</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>2. $A_S$</td>
<td>$\sqrt{2} + 1$</td>
<td>1</td>
<td>-1</td>
<td>$-\sqrt{2} - 1$</td>
<td>$-\sqrt{2} - 1$</td>
<td>-1</td>
<td>1</td>
<td>$\sqrt{2} + 1$</td>
</tr>
<tr>
<td>2. $A_P$</td>
<td>$-\sqrt{2} + 1$</td>
<td>1</td>
<td>-1</td>
<td>$\sqrt{2} - 1$</td>
<td>$\sqrt{2} - 1$</td>
<td>-1</td>
<td>1</td>
<td>$-\sqrt{2} + 1$</td>
</tr>
</tbody>
</table>

#### 2.1.4 Navigation Data Generation

The E5 band signal transmits two signals with distinct navigation data (see section 2.1) and transmission rates [5] [9]:

- E5a-I channel is used for the Open Service and broadcasts data at a rate of 50 symbols/s.
- E5b-I channel is used for integrity checking and broadcasts at a rate of 250 symbols/s.

A 1/2 rate Viterbi convolution coding scheme is used for all data channels, which means that the true data rates in bit/s are half of the symbol rates, in symbols/s. The full description of the data structure is presented in [5]. Since the data transmitted depends on the position of satellites and only the acquisition is studied, in this work it is assumed that the navigation bits are random, neglecting the Galileo message structure.

#### 2.2 AltBOC Modulation

All the signal components presented in the previous sections (the PRN codes, the sub-carriers and the navigation data) are combined to form the Galileo E5 AltBOC signal.

The AltBOC modulation is, in fact, an extension of the BOC modulation, as this scheme uses a sub-carrier signal that adopts a source coding similar to the one used in the BOC scheme [9]. For this reason, in order to get a better comprehension of the AltBOC, a brief discussion of the BOC modulation is presented in section 2.2.1. Furthermore, in section 2.2.2 the modulation that served as the basis of the one used in the E5 band is described. Finally, in section 2.2.3 the modulation used is presented.
2.2.1 BOC modulation

The BOC modulation is used in E1 and E6 transmitted signals. Considering a general baseband signal, $s(t)$, modulated with square waves with nominal power, $P_S$ [9]:

$$s(t) = \sqrt{P_S} e(t), \quad (2.3)$$

with $e(t) \in \{-1, 1\}$.

The $e(t)$ component is, in fact, a combination of the binary navigation data ($d(t)$), with symbol rate $R_D$, and the PRN code ($c(t)$), with chip rate $R_C$ ($R_C > R_D$), as follows:

$$e(t) = d(t) c(t), \quad (2.4)$$

with $d(t), c(t) \in \{-1, 1\}$.

The BOC modulated signal is obtained by multiplying the signal $s(t)$ by a rectangular sub-carrier, illustrated in Figure 2.8, with frequency $f_{\text{sub}}$ usually superior to $R_C = \frac{1}{T_c}$, where $T_c$ is the chip duration. The modulated signal can be expressed as:

$$s_{BOC}(t) = s(t) \cdot \text{sign}[\sin(2\pi f_{\text{sub}} t)]. \quad (2.5)$$

![Figure 2.8: Subcarrier function for BOC modulation.](image)

The modulation chops the chip waveform into smaller truncated square waves, with $f_{\text{sub}}/R_C$ cycles.

In the Galileo literature, the modulation is referred as BOC$(m,n)$, where $m$ is $f_{\text{sub}}$ normalized to the reference 1.023 MHz and $n$ gives the chip rate normalized to the reference 1.023 Mchips/s.

For a bit duration $>> T_c$, the auto-correlation of $s(t)$ is

$$R_S(\tau) \approx \Lambda_{T_c}(\tau), \quad (2.6)$$

where $\Lambda_L(t)$ is the triangle function of duration $2L$, defined as:

$$\Lambda_L(t) = \begin{cases} 
1 - \frac{|t|}{L}, & |t| < L \\
0, & \text{otherwise}
\end{cases}$$

14
The corresponding Power Spectral Density (PSD) can be obtained from Equation (2.6) as:

\[ S(f) = \mathcal{F}\{ R_S(\tau) \} = T_c. \text{sinc}^2(f.T_c). \] (2.7)

The Power Spectral Density of the modulated signal, \( S_{BOC}(f) \), can be approximately expressed as:

\[ S_{BOC}(f) = \alpha S(f - f_{sub}) + \alpha S(f + f_{sub}). \] (2.8)

Since \( s(t) = \pm 1 \) is a random digital signal, \( S(f) \) has the shape of a \( \text{sinc}^2 \) function, as evidenced in Equation (2.6). In Figure 2.9, there is a comparison between the PSD’s of the signal before and after the BOC(1,1) modulation.

![Figure 2.9: PSD of \( s(t) \), before (above) and after (below) the BOC Modulation. [9].](image)

From Figure 2.9, it can be seen that the modulation splits the longer spectrum in two symmetrical parts around the carrier frequency. In addition, there is no power at the central frequency and the main lobes are shifted to positions \( \pm 1.023 \) MHz (equal to \( f_{sub} \)) away from the carrier frequency.

With a proper choice of the \( m \) and \( n \) parameters, a proper spectral separation can be achieved, which means that the same frequency band can be shared by multiple signals, with a minimal signal loss and interference [9] [13].

### 2.2.2 Standard AltBOC modulation

The idea behind the Alternative BOC (AltBOC) is to perform the same operation as in the BOC modulation but using complex rectangular sub-carriers instead.

Considering the same BPSK baseband signal as in the previous section 2.2.1,

\[ s(t) = \sqrt{P_S}.d(t).c(t), \] (2.9)

a new modulation can be achieved by multiplying \( s(t) \) by two rectangular waves that follow the expres-
The complex sub-carriers, \( cr(t) \), can then be defined as:

\[
cr(t) = \text{sign}[\cos(2\pi f_{\text{sub}} t)]
\]

\[
sr(t) = \text{sign}[\sin(2\pi f_{\text{sub}} t)].
\]

(2.10)

These sub-carriers have a similar performance to complex exponentials \( e^{(j2\pi f_{\text{sub}} t)} = \cos(2\pi f_{\text{sub}} t) + j\sin(2\pi f_{\text{sub}} t) \), although, they differ in their spectrum: while the complex exponential's PSD is a delta function centered on \( f_{\text{sub}} \), the sub-carrier \( cr(t) \) PSD is not, and it is shown in Figure 2.10.

![Spectrum of the complex exponential \( cr(t) \) [9]](image-url)

Figure 2.10: Spectrum of the complex exponential \( cr(t) \) [9]

From Figure 2.10, it can be seen that the main peak is around \( f_{\text{sub}} \) (15.345 MHz, in this case), but there is a secondary peak that appears at around \(-3f_{\text{sub}}\). A similar analysis can be made for the sub-carrier \( cr^*(t) \); in this case, the main peak lays around \(-f_{\text{sub}}\) and the secondary one at \(3f_{\text{sub}}\).

The modulation can be achieved by multiplying the BPSK signal \( s(t) \) by one of the sub-carriers \( cr(t) \) or \( cr^*(t) \). The resulting modulation is usually called Complex BOC (CBOC):

\[
s_{\text{CBOC}}(t) = s(t).cr(t).
\]

(2.12)

Analyzing the power spectrum before and after the modulation is applied, Figure 2.11, it is clear that the main peak suffered a frequency shift to \( f_{\text{sub}} \) due to the complex exponential. Additionally, there is an appearance of a secondary peak around \(-3f_{\text{sub}}\). Due to having a lower amplitude, compared to the primary peak, this secondary peak can be neglected.

The Standard AltBOC modulation uses this idea to shift components of the signal to higher or lower frequencies. It transmits over four channels (E5a-I, E5a-Q, E5b-I and E5b-Q) that are shifted in two separate sidebands (E5a and E5b). In each sideband one of the components is transmitted in an in-phase channel while the other is transmitted in a quadrature channel. In detail, each of the components
can be described as:

\[
e_{E5a1}(t) = d_1(t).c_1(t)
\]
\[
e_{E5aQ}(t) = c_2(t)
\]
\[
e_{E5b1}(t) = d_2(t).c_3(t)
\]
\[
e_{E5bQ}(t) = c_4(t),
\]

where \(c_i(t)\) are the orthogonal PRN codes and \(d_i(t)\) the binary navigation data.

The Standard AltBOC can be obtained by combining the components in Equation (2.13), as follows:

\[
s_{S-AltBOC} = [e_{E5a1}(t) + j.e_{E5aQ}(t)].er^*(t)] + [e_{E5b1}(t) + j.e_{E5bQ}(t)].er(t)
\]  

Recalling the same train of thought as done for the CBOC modulation, it can be understood that this modulation shifts the E5a components to a lower frequency and the E5b components to a higher one.

As stated in [14], the result of the modulation can only take 9 values that follow the expression:

\[
s_{S-AltBOC}(t) = A_k.e^{j\frac{k\pi}{4}}
\]

with

\[
A_k = \begin{cases} 
0 & \text{for } k=0 \\
2\sqrt{2} & \text{if } k \text{ is odd} \\
4 & \text{if } k \text{ is even}
\end{cases}
\]

From Equation (2.15) and Figure 2.12, it can be seen that there isn’t a constant power envelope, also it can happen that the power transmitted is null. This is a problem for satellite communications due to the amplifiers needing to work at the saturation level in order to obtain optimal efficiency and minimum distortion.
2.2.3 E5 AltBOC(15,10)

The modulation adopted for the Galileo E5 band is similar to the Standard AltBOC, with some changes in order to achieve a constant power envelope. The sub-carriers were changed from simple square waves to the ones presented in section 2.1.3 and new terms called *product signals* were introduced. The modulation used is called E5 AltBOC(15,10), as it uses a $f_{sub}$ of $15 \times 1.023$ MHz and a $R_C$ of $10 \times 1.023$ MHz.

The E5 signal is formed by the combination of four ranging PRN codes and two data streams, according to Figure 2.13. The signal components are generated according to the following [5]:

- $e_{5a-I}$ from the F/NAV navigation data stream $D_{E5a-I}$ modulated with the unencrypted ranging code $C_{E5a-I}$.
- $e_{5a-Q}$ (pilot component) from the unencrypted ranging code $C_{E5a-Q}$.
- $e_{5b-I}$ from the I/NAV navigation data stream $D_{E5b-I}$ modulated with the unencrypted ranging code.
• $e_{5b-Q}$ (pilot component) from the unencrypted ranging code $C_{E5b-Q}$.

The analytical definition of each one of the components is, as follows [5]:

$$e_{5a-I}(t) = \sum_{i=-\infty}^{\infty} [e_{E5a-I},|i|_E, e_{5a-I},|i|_D] C_{E5a-I} \cdot rect_{C,E5a-I}(t - iT, E5a - I)$$

$$e_{5a-Q}(t) = \sum_{i=-\infty}^{\infty} [e_{E5a-Q},|i|_E, e_{5a-Q},|i|_D] C_{E5a-Q} \cdot rect_{C,E5a-Q}(t - iT, E5a - Q)$$

$$e_{5b-I}(t) = \sum_{i=-\infty}^{\infty} [e_{E5b-I},|i|_E, e_{5b-I},|i|_D] C_{E5b-I} \cdot rect_{C,E5b-I}(t - iT, E5b - I)$$

$$e_{5b-Q}(t) = \sum_{i=-\infty}^{\infty} [e_{E5b-Q},|i|_E, e_{5b-Q},|i|_D] C_{E5b-Q} \cdot rect_{C,E5b-Q}(t - iT, E5b - Q).$$

The wide-band E5 signal is generated with the AltBOC modulation, using the sub-carriers $s_{E5-S}$ and $s_{E5-P}$, discussed in section 2.1.3, and the components defined in Equation (2.16). The result is the signal $s_{E5}$, which follows the expression:

$$s_{E5}(t) = \frac{1}{2\sqrt{2}} (e_{E5a-I}(t) + j e_{E5a-Q}(t)) [s_{E5-S}(t) - j s_{E5-S}(t - T_{S,E5}/4)]$$

$$+ \frac{1}{2\sqrt{2}} (e_{E5b-I}(t) + j e_{E5b-Q}(t)) [s_{E5-S}(t) + j s_{E5-S}(t - T_{S,E5}/4)]$$

$$+ \frac{1}{2\sqrt{2}} (e_{E5a-I}(t) + j e_{E5a-Q}(t)) [s_{E5-P}(t) - j s_{E5-P}(t - T_{S,E5}/4)]$$

$$+ \frac{1}{2\sqrt{2}} (e_{E5b-I}(t) + j e_{E5b-Q}(t)) [s_{E5-P}(t) + j s_{E5-P}(t - T_{S,E5}/4)],$$

where the dashed signal components, $\bar{e}_{E5a-I}$, $\bar{e}_{E5a-Q}$, $\bar{e}_{E5b-I}$ and $\bar{e}_{E5b-Q}$, represent the product signals:

$$\bar{e}_{E5a-I} = e_{E5a-Q} . e_{E5b-I}$$

$$\bar{e}_{E5a-Q} = e_{E5a-I} . e_{E5b-Q}$$

$$\bar{e}_{E5b-I} = e_{E5a-I} . e_{E5a-Q}$$

$$\bar{e}_{E5b-Q} = e_{E5a-I} . e_{E5b-Q}.$$ 

The purpose of these product signals is solely to maintain a constant power envelope. Also, they don’t carry any useful information and, therefore, are usually neglected when the signals are handled.

As seen in equation 2.17, the signal components are multiplied by “complex exponentials”: $s_{E5-S}(t) = j s_{E5-S}(t - T_{S,E5}/4)$ for the E5a sideband and $s_{E5-S}(t) + j s_{E5-S}(t - T_{S,E5}/4)$ for the E5b sideband.

The PSD of the second exponential is represented in Figure 2.14. It is similar to the PSD of the first one: they are symmetric in relation to the zero frequency shift.

Using the same train of thought as in the description of the BOC and CBOC modulations, it can be concluded that the complex exponentials perform a frequency shift on the $sinc^2$ PSD functions of the signal components. In detail, the first “exponential” shifts the E5a-I and E5a-Q components to the E5a
band and the second one shifts the E5b-I and E5b-Q to the E5b band.

The PSD of the full AltBOC(15,10) modulation can be expressed as [15]:

$$G_{AltBOC(15,10)}(f) = T_c \cos(\pi f T_c) \left[ 4 \sin^2\left(\frac{f T_c}{12}\right) \cos\left(\frac{f T_c}{6}\right) - \sin^2\left(\frac{f T_c}{3}\right) \cos\left(\frac{\pi f T_c}{6}\right) \right]$$ (2.19)

From Equation (2.19) it is possible to plot the PSD of the AltBOC modulation, which is presented in Figure 2.15.

In Figure 2.15, it can be seen that the main lobes of the AltBOC(15,10) modulation span over 50MHz. This means that a receiver capable of handling this signal needs to be able to operate with large bandwidths.

A peculiar aspect about the AltBOC is its effect on the auto-correlation function (ACF). According to
[15], the modulation’s ACF can be expressed as:

\[
R_{\text{AltBOC}(15,10)}(\tau) = 8\Lambda T_c/6(\tau) - \frac{16}{3} \Lambda T_c/6(|\tau| - \frac{T_c}{3}) + \frac{8}{3} \Lambda T_c/6(|\tau| - \frac{2T_c}{3}) \\
- \frac{1}{3} \Lambda T_c/12(|\tau| - \frac{T_c}{12}) - \frac{1}{3} \Lambda T_c/12(|\tau| - \frac{3T_c}{12}) + \frac{1}{3} \Lambda T_c/12(|\tau| - \frac{5T_c}{12}) \\
+ \frac{1}{3} \Lambda T_c/12(|\tau| - \frac{7T_c}{12}) - \frac{1}{3} \Lambda T_c/12(|\tau| - \frac{9T_c}{12}) - \frac{1}{3} \Lambda T_c/12(|\tau| - \frac{11T_c}{12})
\] (2.20)

Figure 2.16 shows the plot of the ACF made from Equation (2.20).

Figure 2.16: Normalized ACF of the AltBOC(15,10) signal.

By comparing Figure 2.16 with Figure 2.17, it can be seen that the ACF of the AltBOC presents a narrow correlation peak compared to the BPSK’s ACF. However, it also presents secondary peaks with considerable amplitude (around 70% of the power of the main peak) that, if not handled appropriately, can lead to false locks in the acquisition and tracking stages, which degrade the performance of the complete Galileo receiver.

Figure 2.17: Normalized ACF of the BPSK signal.
Equivalent Modulation Type

The rather complex modulation scheme presented can be implemented using a simple Look-Up Table (LUT) to map the phase assignments.

As expressed in [5], the E5 AltBOC modulation can be described as an 8-PSK signal. Therefore, the modulated signal can, alternatively, be expressed as:

\[ s_{E5}(t) = e^{j \frac{\pi}{4} k(t)} \quad \text{with} \quad k \in \{1, 2, 3, 4, 5, 6, 7, 8\}, \]  \hspace{1cm} (2.21)

where \( k(t) \) defines the scattered plot number in Figure 2.18.

![Figure 2.18: 8-PSK Phase-State Diagram of E5 AltBOC Signal [5].](image)

The idea is to allocate any of the eight sub-carrier phase states and any of the sixteen \( 2^4 \) different possible states of the quadruple \( e_{5a-I}(t), e_{5a-Q}(t), e_{5b-I}(t) \) and \( e_{5b-Q}(t) \) to a phase spot in the constellation, using a LUT with 128 (8x16) different entries. The value of the constellation spot depends also on time. Therefore, time is partitioned first in sub-carrier intervals, \( T_{S,E5} \) and further sub-divided in eight equal sub-periods. The index \( i_{T_s} \) of the actual sub-period is given by

\[ i_{T_s} = \text{integer part} \left[ \frac{8}{T_{S,E5}}(t \text{ modulo } T_{S,E5}) \right]. \]  \hspace{1cm} (2.22)

Since the LUT is composed only by 128 entries, this efficient technique based on a LUT will be used for the signal generation sections on board of the Galileo satellites, obtaining a significant simplification for the necessary hardware and computational burden [9]. The complete LUT is represented in Table 2.6.
2.3 Receiver Hardware Considerations

Even though the purpose of this work is to study the E5 signal acquisition using software methods, a minimum hardware is always needed. Therefore, the hardware can not be neglected and, in this chapter, the fundamental hardware needed to achieve a correct acquisition is presented. There are shown possible configurations for three types of receiver [9]:

- Single Band receiver (section 2.3.1): this receiver only processes one of the E5 sub-bands (E5a or E5b).
- Separate Double Band receiver (section 2.3.2): this receiver processes each one of the sub-bands individually and combines the results in the software processing section.
- Coherent Double Band receiver (section 2.3.3): this receiver processes the full-band AltBOC signal.

### 2.3.1 Single Band Receiver

The single band receiver has the simplest architecture and is appropriate for simple low-cost receivers. In Figure 2.19 is represented a possible schematic of this type of receiver.

The first section, corresponding to the RF front end, consists of an antenna, an amplifier and a filter. The antenna needs to cover a wide spatial angle to receive the maximum number of signals and needs to be designed in order to work correctly in the E5 band (1164-1215 MHz). The amplifier needs to be used because the power of the received signal is low. A band-pass filter (centered at the chosen sideband, E5a or E5b, central frequency with a bandwidth of about 20 MHz) is used to “separate” the sub-band signal from the rest of the spectrum [12].

The next part, corresponding to the IF (Intermediate Frequency) section, usually has an oscillator, a mixer, two filters, one or more amplifiers and an analog to digital converter (ADC). The oscillator and the

<table>
<thead>
<tr>
<th>$t'$</th>
<th>$t$</th>
<th>$i$, $j$, $s$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0, $T_{S,E5}/8$</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1, $T_{S,E5}/8$</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>2, $T_{S,E5}/8$</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>3, $T_{S,E5}/8$</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>4, $T_{S,E5}/8$</td>
</tr>
</tbody>
</table>

Table 2.6: Look-up Table for AltBOC Phase States [5].
mixer are used to down-convert the signal to an IF frequency. The amplifiers are designed to achieve a signal level high enough that all levels of the ADC are activated, but also low enough that no part of the ADC becomes saturated. The filter following the amplifiers is used to reduce the noise introduced by the amplifiers. Finally, the ADC quantizes and samples the analog signal in order to achieve a digital signal that can be processed in software.

The last stage of the receiver, the processing, corresponds to the software part, whose acquisition stage will be discussed in the following chapters.

### 2.3.2 Separate Double Band Receiver

The separate double band receiver allows the processing of both the sidebands individually. To achieve this, the receiver requires significantly more hardware but offers slightly better performance than the previous one. The architecture is presented in Figure 2.20.

As Figure 2.20 shows, this receiver is an extension of the single band receiver. The main differences are the need of two filters after the antenna, to isolate each one of the sidebands, and the need of two IF sections (one for each sideband). For each sideband, the functional blocks of the IF section are the same as in the single band receiver.

The main advantage of this receiver is the possibility to correct the ionospheric error. Since the receiver works at two different frequency bands, it can estimate the error based on the delay between...
the signals received in each sideband. However, better accuracy results from using two bands with a larger frequency separation: for instance the E5a and E1 bands.

### 2.3.3 Coherent Double Band receiver

The coherent double band receiver is the one that can achieve the best performance out of the presented receiver configurations. The hardware of the receiver is presented in Figure 2.21.

![Figure 2.21: Architecture of a Coherent Double Band Receiver.](image)

In terms of hardware, this receiver is similar to the single band receiver. The only difference, in terms of hardware, resides in the entry filter: in the coherent double band receiver, the filter bandwidth must be large enough ($\geq 51$ MHz) to accommodate the full E5 band. The main difference between the receivers is in the software processing section. This receiver allows the use of the most complex techniques that can achieve the best results.

### 2.3.4 Hardware Parameters

#### Intermediate Frequency

As shown in section 2.3.1, an oscillator and a mixer are used to convert the received signal with frequency $f_{RF}$ into a signal with a smaller frequency, called intermediate frequency $f_{IF}$. This process can be explained as follows, where the RF signal $s_{RF}$ is a common sinusoidal function:

$$s_{IF}(t) = s_{RF}(t).s_{osc}(t)$$

$$= x(t).\cos(2\pi f_{RF}t).\cos(2\pi f_{osc}t)$$

$$= \frac{x(t)}{2}[\cos(2\pi(f_{RF} - f_{osc})t) + \cos(2\pi(f_{RF} + f_{osc})t)], \quad (2.23)$$

where $x(t)$ is the baseband signal and $f_{osc}$ is the frequency of the oscillator signal. By applying a low-pass filter to $s_{IF}$, the high frequency component can be eliminated and the result is a signal with frequency $f_{IF} = (f_{RF} - f_{osc})$:

$$s_{IF}(t) = \frac{x(t)}{2}[\cos(2\pi(f_{RF} - f_{osc})t)]. \quad (2.24)$$

This down-conversion is made because working with high frequencies such as the ones used in the
E5 band lead to some issues: the amplifiers are difficult to build and therefore expensive; and an ADC, capable of having an input bandwidth to accommodate the high input frequency, is extremely hard to build and has fewer effective bits [12]. Although this process is exemplified by a single stage heterodyne, the down-conversion could be performed in two or more stages, shifting the central frequency of the received signal in more steps. The only difference is the receiver implementation (different cost and complexity of the required components, different noise performances) [9].

The value of this intermediate frequency in conventional GNSS is typically in the order of the Megahertz (MHz). The RF signal is not down-converted to the zero center frequency for the following reasons:

- Ambiguity in the determination of the Doppler frequency. If the $f_{IF}$ is close to zero, the acquisition stage cannot distinguish between a negative and a positive Doppler shift. So the $f_{IF}$ chosen must be greater than the foreseen maximum value of the Doppler shift (typically 5 kHz for a fixed receiver).

- Overlapping of the input signal spectrum. If the IF is not high enough, the positive part of the spectrum may overlap with its corresponding negative part, which results in interference and signal degradation. So the $f_{IF}$ chosen must be greater than half of the input signal bandwidth, which corresponds to 10 MHz in the single band and separate double band receivers and 20.5 MHz in the coherent double band receiver.

### Sampling Frequency

The sampling frequency (also designated sampling rate), $f_s$, is specified by the ADC. The choice of the sampling rate depends on the type of receiver user. There are two main factors that influence the choice of the sample frequency: the first is the input signal bandwidth and the second is the correlation properties of the input signal modulation.

If the receiver processes a sideband as a BPSK signal separately from the other sideband, the choice of sampling rate must take into account the correlation properties of the BPSK correlation. This is the case of the single band receiver and the separate double band receiver.

![Normalized ACF of the BPSK modulation](image)

**Figure 2.22:** Normalized ACF of the BPSK modulation.

As Figure 2.22 shows, a 50% loss of power relatively to the maximum correlation, occurs for a time delay of half of a primary chip period. So, to guarantee that there is always a correlation value of at least

26
half the maximum correlation, the sampling rate must be at least double the primary code frequency, which is $2 \times 10.230 = 20.460$ MHz. In addition, the Nyquist theorem states that, to avoid ambiguities in the frequency domain, the sampling frequency needs to be greater or equal to twice the input signal bandwidth [12]. Since, in this case, the input bandwidth is equal to 20 MHz, it means that the sampling rate for these receivers needs to be at least 40 MHz.

When the coherent double band receiver is used, a higher sampling frequency is required. This is due to the narrower correlation peak and the existence of secondary peaks, that, if not treated correctly, may lead to a false lock in the acquisition step and to the higher input bandwidth.

Looking at Figure 2.23, it can be seen that the power loss of 3dB, in the main peak relatively to the maximum correlation, occurs for a time delay of $0.077 \times T_c$. So, similarly to what was done before, in order to guarantee that there is always a correlation value of at least half the maximum correlation, the sampling rate must be of at least $1/0.077 \times R_c \approx 133$ MHz. Since the input bandwidth is approximately 51 MHz, the Nyquist theorem states that a minimum sampling frequency of 102 MHz is necessary. In this case, the limiting factor is the correlation properties of the AltBOC modulation, so the sampling frequency must be greater or equal to 133 MHz.
Chapter 3

Acquisition Concepts

To continuously determine its position, a GNSS receiver typically needs to perform four steps: acquisition, tracking, data demodulation and position determination. As the purpose of this thesis is the study of the acquisition stage, this is the only step that is in-depth described.

- **Acquisition.** The acquisition process is the first step any GNSS receiver must do, in order to take advantage of the satellite system capabilities. In this step, it is determined if a signal from a given satellite is or not present. If a signal is considered to be present, the relevant parameters for the next steps of the receiver are determined. Those parameters are the beginning of the primary and secondary codes and the carrier frequency of the input signal. Even though the signals are transmitted with a fixed carrier frequency, because the receiver is in motion in relation to the transmitter satellite, a Doppler frequency shift is usually introduced in the signal. The code alignment and the Doppler frequency estimations are crucial to synchronize the Delay-Locked Loop (DLL) and the Phase-Locked Loop (PLL), respectively, during the tracking stage.

- **Tracking.** Signal tracking consists on following a satellite signal by constantly aligning a local signal replica with the received signal using two closed loops: DLL and PLL (or FLL). The objective of the tracking process is finding the Doppler frequency associated with the receiver-transmitter relative movement and code delay resulting from the distance traveled by the signal [16].

- **Data Demodulation.** Once the code delay and Doppler frequency shift are determined, the navigation bits can be recovered from the correlator outputs. Once the full navigation message is read, the receiver translates the navigation bits into navigation data, based on the message structure defined in [5].

- **Position Determination.** Finally, after the navigation data is found, the receiver determines its position based on the difference of times between the transmission and reception of the signals from multiple satellites, as described in section 1.2.

In this section, the basics of the acquisition stage are presented. At first, the concepts of correlation (section 3.1), search space (section 3.2) and detection (section 3.3) are introduced. At the end,
two methods of combining the correlation results to obtain a better signal-to-noise ratio are discussed (sections 3.4 and 3.5).

### 3.1 Correlation

When a Single Band receiver\(^1\) (section 2.3.1) designed for the E5a sideband is used, the received signal can be expressed as \[17\]:

\[
r(t) = A \cdot [Re\{s_{E5-a}(t - \tau)\} \cos((\omega_{IF} + \omega_d)t + \phi) - Im\{s_{E5-a}(t - \tau)\} \sin((\omega_{IF} + \omega_d)t + \phi)] + n_w(t), \tag{3.1}
\]

where \(s_{E5-a}\) are the E5a sideband components of the baseband AltBOC signal described in section 2.2.3, \(\omega_{IF}\) is the intermediate frequency described in section 2.3.4, \(\omega_d\) is the Doppler shift, \(\tau\) is the time delay with respect to the transmitted signal, \(\phi\) is the phase of the received signal and \(n_w\) is additive white Gaussian noise with power spectral density \(N_0/2\). Equation (3.1) can be approximated by \[17\]

\[
r(t) \approx \frac{A \cdot \sqrt{2}}{\pi} [e_{E5a-I}(t - \tau) \cos((\omega_{IF} + \hat{\omega}_d - \omega_s)t + \phi) - e_{E5a-Q}(t - \tau) \sin((\omega_{IF} + \hat{\omega}_d - \omega_s)t + \phi)] + n_w(t), \tag{3.2}
\]

where \(\omega_s\) is the frequency of the sub-carrier (section 2.1.3).

In order to obtain a correlation value from the received signal, it is necessary to perform two steps: a carrier wipe-off and a code wipe-off. The order of these operations is interchangeable, but, for a better comprehension in this section, the wipe-offs are described by performing a carrier wipe-off before the code wipe-off.

The carrier wipe-off, as the name indicates, consists in eliminating the modulation by the carrier frequency. This process is exemplified in Figure 3.1.

![Figure 3.1: Carrier Wipe-off.](image)

As seen in Figure 3.1, the received signal is multiplied by two reference signals in order to obtain an in-phase and a quadrature signals\(^2\). Due to the existence of a phase error, \(\phi\), between the received signal and the signals generated by the local oscillator \(\cos((\omega_{IF} + \hat{\omega}_d - \omega_s)t)\) and \(\sin((\omega_{IF} + \hat{\omega}_d - \omega_s)t)\), to eliminate the need for a phase error search, it is common to analyze both signals during this step.

\(^1\)The signal received in each one of the branches of a Separate Double Band receiver (section 2.3.2) has the same expression as the signal received in a Single Band receiver.

\(^2\)These in-phase and quadrature signals do not correspond exactly to the in-phase and quadrature signals of the E5 modulation because there is a phase difference.
After low pass filtering the signals can be expressed as:

\[ r_I(t) = \frac{A}{\pi} [e^{E_{5a-I}(t-\tau)} \cos((\omega_d - \tilde{\omega}_d)t + \phi) + e^{E_{5a-Q}(t-\tau)} \sin((\omega_d - \tilde{\omega}_d)t + \phi)] + \tilde{n}_{w_I}(t) \]

\[ r_Q(t) = \frac{A}{\pi} [-e^{E_{5a-I}(t-\tau)} \sin((\omega_d - \tilde{\omega}_d)t + \phi) + e^{E_{5a-Q}(t-\tau)} \cos((\omega_d - \tilde{\omega}_d)t + \phi)] + \tilde{n}_{w_Q}(t), \tag{3.3} \]

where \( \tilde{n}_{w_I}(t) \) and \( \tilde{n}_{w_Q}(t) \) are independent low pass Gaussian noises with power spectral density \( N_0 \) over the filters bandwidth.

From Equation (3.3), it can be seen that the carrier has been wiped, as the frequency of the processed signals is equal to the difference between the actual Doppler shift and the tested Doppler shift \( \tilde{\omega}_d \).

The code wipe-off is performed by multiplying each one of the in-phase and quadrature signals by a local code replica\(^3\) with an estimated code delay \( \tilde{\tau} \), as Figure 3.2 shows.

\[ r_I(t) \approx \frac{A}{\pi} [e^{E_{5a-Q}(t-\tau)} \cdot c^{E_{5a-Q}}(t-\tilde{\tau}) \sin((\omega_d - \tilde{\omega}_d)t + \phi)] + \tilde{n}_{w_I}(t) \]

\[ r_Q(t) \approx \frac{A}{\pi} [-e^{E_{5a-Q}(t-\tau)} \cdot c^{E_{5a-Q}}(t-\tilde{\tau}) \cos((\omega_d - \tilde{\omega}_d)t + \phi)] + \tilde{n}_{w_Q}(t). \tag{3.4} \]

After both the wipe-off are complete, a correlation value is obtained for each one of the signals by integrating the results over a period of \( T_{\text{int}} \), also known as dwell time. The outputs are as expressed:

\[ S_I = \frac{A}{\pi} \int_0^{T_{\text{int}}} [e^{E_{5a-Q}(t-\tau)} \cdot c^{E_{5a-Q}}(t-\tilde{\tau}) \sin((\omega_d - \tilde{\omega}_d)t + \phi) + \tilde{n}_{w_I}(t)]d(t) \]

\[ S_Q = \frac{A}{\pi} \int_0^{T_{\text{int}}} [e^{E_{5a-Q}(t-\tau)} \cdot c^{E_{5a-Q}}(t-\tilde{\tau}) \cos((\omega_d - \tilde{\omega}_d)t + \phi) + \tilde{n}_{w_Q}(t)]d(t) \tag{3.5} \]

For \( \tilde{\omega}_d \approx \omega_d \), Equation (3.5) can be simplified to

\[ S_I \approx \frac{A}{\pi} T_{\text{int}} \cdot \sin \phi \cdot R_{E_{5a-Q}}(\tilde{\tau} - \tau) + N_I \]

\[ S_Q \approx \frac{A}{\pi} T_{\text{int}} \cdot \cos \phi \cdot R_{E_{5a-Q}}(\tilde{\tau} - \tau) + N_Q, \tag{3.6} \]

where \( R_{E_{5a-Q}}(\tau) \) is the autocorrelation function of the primary code \( c^{E_{5a-Q}}(t) \) and \( N_I \) and \( N_Q \) are Gaussian noises.

\(^3\)The process is exemplified by using the E5a-Q code, but it is also true for the E5a-I code.
When $\hat{\tau} \approx \tau$, the results of Equation (3.5) can be simplified to

$$S_I \approx \frac{A}{\pi} \int_0^{T_{int}} \sin[(\omega_d - \hat{\omega}_d)t + \phi]dt + N_I$$

$$= \frac{AT_{int}}{\pi} \cdot \text{sinc}[(f_d - \hat{f}_d)T_{int}] \sin[\omega_d - \hat{\omega}_d] \frac{T_{int}}{2} + \phi + N_I$$

(3.7)

$$S_Q \approx \frac{A}{\pi} \int_0^{T_{int}} \cos[(\omega_d - \hat{\omega}_d)t + \phi]dt + N_Q$$

$$= \frac{AT_{int}}{\pi} \cdot \text{sinc}[(f_d - \hat{f}_d)T_{int}] \cos[\omega_d - \hat{\omega}_d] \frac{T_{int}}{2} + \phi + N_Q.$$}

If no special techniques are applied, this integration time is limited to the occurrence of bit transitions, outside of the primary code. These transitions are caused by the secondary code bits, whose duration is 1ms. So, to avoid bit transitions, the typical dwell time is 1ms.

The correlation results can then be combined, in an envelope detector, to obtain the final correlation value:

$$S^2 = S_I^2 + S_Q^2$$

(3.8)

For instance, neglecting the effect of noise in Equation (3.6) leads to

$$S^2 = \frac{A^2}{\pi^2} T_{int}^2 R_{E5a-\hat{Q}}^2(\hat{\tau} - \tau),$$

(3.9)

which is plotted in Figure 3.3.

![Figure 3.3: Output of the Envelope Detector for $\hat{\omega}_d = \omega_d$.](image)

Doing the same process for Equation (3.7) yields

$$S^2 = \frac{A^2}{\pi^2} T_{int}^2 \text{sinc}^2[(f_d - \hat{f}_d)T_{int}],$$

(3.10)

which is plotted in Figure 3.4.

If the acquisition is to be performed for the full AltBOC modulation (only possible in the coherent double band receiver), the correlation process is similar to what was presented. The biggest difference resides on the code wipe-off. Instead of performing the wipe-off with a single code replica, a replica of the modulation (codes for any of the four channels plus the sub-carrier) needs to be generated. This is
usually done making use of the LUT implementation, previously discussed in section 2.2.3. More details are presented in section 4.5.

### 3.2 Search Space

The acquisition step can be described as a two dimension search: in one dimension the time uncertainty ($\tau$) is searched and in the other the frequency uncertainty ($f_d$), as shown in Figure 3.5.

During the acquisition stage, a correlation is performed for every single cell\(^4\) of the search space. The size of the search space depends on the time and frequency increments. In general, for smaller increments, a better resolution and sensibility can be achieved but the size of the search increases, which means slower acquisition times. On the other hand, bigger increments allow a faster acquisition but with worse performance. The two increments that define the search space are:

- Doppler frequency: the range of the search is usually $\pm 5\text{kHz}$ when the receiver is stationary or in a slow movement, but can reach $\pm 10\text{kHz}$ when the receiver is in a fast movement. The resolution that can be achieved is dependent on the integration time $T_{int}$ and follows the expression [13]:

\[^{4}\text{A cell is a combination of a given code and frequency bins.}\]
Increment = \frac{1}{2T_{int}} \text{ (see also Figure 3.4).}

- Code delay: the range of the search correspond to one entire primary code period, which is 1ms. The increment is typically made equal to \frac{1}{2}T_c [18] (see also Figure 3.3): Increment = \frac{T_c}{2}.

The size of the full search space is equal to the number of code bins times the number of frequency bins, which means that

\[ size = 4 \frac{T_{int} \text{ range}_f \text{ range}_c}{T_c}, \]

where \text{range}_f and \text{range}_c correspond to the Doppler frequency and code delay search ranges, respectively.

### 3.3 Detection

After the correlation values are calculated for every single one of the search cells, the receiver needs to be able to determine if the signal from a given satellite is, or not, present. To achieve this, the receiver compares the maximum correlation value with a threshold value. If the correlation value is higher than the threshold, the signal is considered present, otherwise it is considered missing.

There are two types of methods to determine the threshold [12]. The first approach uses simulated results to determine the threshold. The biggest advantage of this method is that the noise of the collected data needs to be calculated only once, and the same result will always be used as the threshold. The biggest disadvantage is that if the noise conditions change, the receiver may not be able to perform the detection.

The second approach, and the one used in this work, uses real data and determines the threshold experimentally. To perform this, it is first necessary to determine the carrier-to-noise-density ratio (C/N₀).

#### 3.3.1 C/N₀ Estimation

The most common and well-known method to estimate the C/N₀ in GNSS is the Narrow-Wideband Power Ratio (NWPR) method, described in [19]. This method evaluates the total power of the correlation process at two different noise bandwidths: a Wide-Band Power (WBP_k) measurement taken over the noise bandwidth \( \frac{1}{T_{int}} \); and a Narrow-Band Power (NBP_k) measurement taken over the noise bandwidth \( \frac{1}{MT_{int}} \), where \( M \) is a positive integer [20]. These two measurements can be expressed as:

\[
WBP_k = \left( \sum_{i=1}^{M} S^2_{I_i} + S^2_{Q_i} \right)_k,
\]

\[
NBP_k = \left( \sum_{i=1}^{M} S^2_{I_i} \right)_k + \left( \sum_{i=1}^{M} S^2_{Q_i} \right)_k.
\]

The average ratio of the narrowband to wideband power metrics gives an estimate of the noise power,
\( \bar{\mu}_{NP} \), by the following expressing:

\[
\bar{\mu}_{NP} = \frac{M}{N} \sum_{k=0}^{N/M-1} \frac{NBP_k}{WB_P_k},
\]

(3.13)

where \( N \) is the total number of samples used to estimate the \( C/N_0 \).

The measurement in Equation (3.13) can be used in the \( C/N_0 \) estimator:

\[
\hat{C}/N_0 = \frac{1}{T_{int}} \frac{\bar{\mu}_{NP} - 1}{M - \bar{\mu}_{NP}}.
\]

(3.14)

In [21], an alternative method to estimate the carrier-to-noise-density ratio is presented. It is called Modified Maximum Likelihood (MML) estimator. Let \( \vec{S} = (\vec{S}_1, ..., \vec{S}_M) \) represent a vector of \( M \) consecutive complex correlator outputs, as \( \vec{S}_1 = S_I + jS_Q \). This estimator determines the SNR of the incoming signal using the observation vector \( \vec{S} \) and later converts the SNR estimate into \( C/N_0 \), using

\[
SNR = \frac{C}{N_0} T_{cs},
\]

(3.15)

where \( T_{cs} \) corresponds to the duration of a secondary code bit.

The probability density function of the correlator outputs can be written as [21]:

\[
f(\vec{S}, C, \sigma^2) = \frac{1}{(2\pi \sigma^2)^M} \exp \left( \frac{1}{2\sigma^2} \sum_{m=1}^{M} |\vec{S}_m - \sqrt{\frac{C}{2}} \exp(j\psi)|^2 \right),
\]

(3.16)

where \( C \) is the carrier power, \( \sigma^2 \) the noise variance and \( \psi \) represents the net phase difference between the incoming signal and the local replica.

The carrier power can be estimated by equating the partial derivative of the log likelihood function, \( ln[f(\vec{S}, C, \sigma^2)] \), in order to \( C \), to zero which gives

\[
\hat{C} = \left| \frac{1}{M} \sum_{m=1}^{M} \vec{S}_m \right|.
\]

(3.17)

The noise variance of the correlator output can be estimated using the observation vector \( \vec{S}_{ext} \). These complex correlator outputs can be obtained by correlating the incoming signal with a non-existing PRN code (not being transmitted by any satellite). Since [5] defines 50 different codes and only 36 are assigned to the satellites, this can be achieved by using any of the last 14 PRN codes (37-50). The noise variance estimation is given by [21]

\[
\sigma_n^2 = \frac{1}{2(M - 1)} \sum_{i=1}^{M} \left| \vec{S}_{ext,i} - \frac{1}{M} \sum_{k=1}^{M} \vec{S}_{ext,k} \right|^2.
\]

(3.18)

The SNR estimate is given by the ratio between the carrier power and noise variance:

\[
SNR = \frac{\left| \frac{1}{M} \sum_{m=1}^{M} \vec{S}_m \right|^2}{\frac{1}{2(M - 1)} \sum_{i=1}^{M} \left| \vec{S}_{ext,i} - \frac{1}{M} \sum_{k=1}^{M} \vec{S}_{ext,k} \right|^2},
\]

(3.19)
which leads to
\[
C/N_0 = \frac{\frac{1}{M} \sum_{m=1}^{M} S_m}{\frac{2T_{cs}}{2(M-1)} \sum_{i=1}^{M} |S_{ext,i} - \frac{1}{M} \sum_{k=1}^{M} S_{ext,k}|}^2 .
\]  
(3.20)

### 3.3.2 Threshold Setting

Naming \( f_{na}(x) \) the probability density function (PDF) at the envelope detector output when there is no signal alignment, i.e. a satellite signal is not present, the false alarm probability is given by

\[
P_{fa} = \int_{V_t}^{+\infty} f_{na}(x) \, dx, \quad (3.21)
\]

where \( V_t \) corresponds to the threshold value.

The same way, naming \( f_a(x) \) the pdf at the envelope detector output when a satellite signal is present, the probability of correct detection is given by

\[
P_d = \int_{V_t}^{+\infty} f_a(x) \, dx. \quad (3.22)
\]

Recalling Equation (3.5), the noise correlation outputs are given by

\[
N_I = \int_0^{T_{int}} \tilde{n}_{w_I}(t) \, dt,
\]
\[
N_Q = \int_0^{T_{int}} \tilde{n}_{w_Q}(t) \, dt. \quad (3.23)
\]

Assuming that the filters in Figure 3.1 have a large bandwidth, the power spectral density of \( \tilde{n}_{w_I}(t) \) and \( \tilde{n}_{w_Q}(t) \) are given approximately by \( N_0 \). This result is also true for noises \( \tilde{n}_{w_I}(t) \) and \( \tilde{n}_{w_Q}(t) \). Thus, both \( S_I \) and \( S_Q \) are random Gaussian processes with zero mean and variances equal to

\[
\sigma^2_{S_I} = \sigma^2_{S_Q} = \int_0^{T_{int}} \int_0^{T_{int}} E\{ \tilde{n}_{w_I}(t) \tilde{n}_{w_Q}(\tau) \} \, dt \, d\tau = N_0 T_{int}. \quad (3.24)
\]

If the local generated code is not aligned with the received signal, the integrals in equation 3.5 are approximately equal to zero, except on the noise parts, and the quantity \( S^2 \) (equation 3.8) has a Rayleigh distribution. The PDF \( f_{na}(x) \) is then given by

\[
f_{na}(x) = \frac{x}{N_0 T_{int}} e^{-\frac{x^2}{2N_0 T_{int}}} u(x), \quad (3.25)
\]

where

\[
u(x) = \begin{cases} 
1, & x \geq 0 \\
0, & x < 0
\end{cases}
\]

To set the threshold limit, a value for the false alarm probability is chosen, and the corresponding threshold is set on the basis of the PDF expressed in Equation (3.25). Recalling Equation (3.21), the
false alarm probability is given by

\[ P_{fa}(V_t) = \int_{-\infty}^{+\infty} \frac{x}{N_0 \cdot T_{int}} e^{-\frac{x^2}{2N_0 \cdot T_{int}}} \, dx \]

\[ = e^{-\frac{V_t^2}{2N_0 \cdot T_{int}}} \]  

(3.26)

The threshold can then be estimated from Equation (3.26) for a chosen probability of false alarm as:

\[ V_t = \sqrt{-2N_0 \cdot T_{int} \ln(P_{fa})} \]  

(3.27)

The noise power spectral density needs to be estimated, and that can be done using the methods presented in section 3.3.1.

### 3.4 Coherent Integration

In conventional GNSS receivers, it is common to perform a coherent integration over several primary code periods, in order to obtain a better SNR and therefore a stronger correlation peak.

The coherent integration consists on performing summations in a way that different correlation samples are averaged. If more correlations samples are considered, because the noise has zero mean, the noise influence on the correlation is reduced, while the signal contribution is increased. Therefore, the correlation value of the samples where only noise is present lowers and in the ones where signal is present increases [9].

However the use of large coherent integration times is not possible due to some negative effects:

- Doppler shift. Since the Doppler frequency shifts over time, if the integration time is long enough, some samples can interfere with another.

- Bit transitions. If a bit transition, caused by the navigation data and the secondary code, occurs during the integration window, the correlation values are reduced.

- Computational burden. If the integration time increases, the amount of data that needs to be processed also increases, which leads to higher acquisition times.

In conclusion, for the acquisition of E5 signals, the coherent integration time can not exceed 1ms to avoid the bit transitions caused by the secondary code. However, in section 4.3.5 a method is introduced, which allows to use larger integration times.

### 3.5 Non-coherent Integration

Another way to improve the correlation output is to use non-coherent integrations. This integration differs from the one presented in the previous section, as it sums the results after the envelope detector. This means that not only the signal power is increased but also the noise power.
When compared to the coherent integration, the non-coherent integration the gained SNR is smaller, and therefore less effective, because the envelope detector is not zero mean. In fact, according to [9], doubling the integration time the SNR gain is $\sqrt{2}$ times superior for the coherent integration. However, this integration does not suffer the secondary bit transition issue, so a larger integration time can be used.
Chapter 4

Acquisition Methods for the E5 Band

In the present chapter, the categorization of the various acquisition methods based on which signals are acquired (section 4.1) and on how the search space is searched (section 4.2) is presented. Then, the most suitable acquisition methods for the E5 band are described, being shown examples for each category of acquisition. At the end, in section 4.6, the procedures necessary to transition from the acquisition stage to the tracking one are explained.

4.1 Categorization of the Acquisition Methods

Because of the characteristics of the E5 AltBOC signal (see chapter 2), there are multiple ways the acquisition can be processed. Below, the most common methods for a search strategy based acquisition are presented. It is assumed that the secondary code phase is unknown and the aim of the acquisition is to acquire only the primary code [8].

4.1.1 Single Sideband Acquisition (SSB)

This method, commonly used in single band receivers (section 2.3.1), only uses one of the sidebands to perform the acquisition. The input IF signal is mixed with the local carrier centered at one of the sidebands and the result is then mixed with the local signal void of any sub-carrier. There are two variants:

- Using any of the E5a-I, E5a-Q, E5b-I or E5b-Q channels with a coherent integration limit of 1ms (one secondary code chip duration).

- Using a non-coherent integration of {E5a-I, E5a-Q} or {E5b-I, E5b-Q}. The integration duration is not constrained by the spreading codes or the data (but by the receiver clock and user dynamics).

A coherent combining of pilot and data channels is not directly possible, since the secondary codes used in each channel are different from each other.
4.1.2 Double Sideband Acquisition (DSB)

This method is usually used in separate double band receivers (section 2.3.2). Here the correlation results are calculated individually for each sideband and then non-coherently combined. The two variants are:

- Using a non-coherent integration of \{E5a-Q, E5b-Q\} or \{E5a-I, E5b-I\}: combination of pilot channels or data channels.
- Using a non-coherent integration of \{E5a, E5b\}.

4.1.3 Full-band Independent Code Acquisition (FIC)

In this method, that can only be used for a coherent double band receiver, the local carrier is centered at the center of the E5 band. The reference signal contains the spreading code and the sub-carrier corresponding to the signal components of interest. It can be applied in four different ways:

- Using any of the E5a-I, E5a-Q, E5b-I or E5b-Q channels with a coherent integration limit of 1ms (one secondary code chip duration).
- Using a non-coherent integration of \{E5a-I, E5a-Q\} or \{E5b-I, E5b-Q\}. The integration duration is not constrained by the spreading codes or the data (but by the receiver clock and user dynamics).
- Using a non-coherent integration of \{E5a-Q, E5b-Q\} or \{E5a-I, E5b-I\}: combination of pilot channels or data channels.
- Using a non-coherent integration of \{E5a, E5b\}.

In a FIC method, a locally generated individual code with the corresponding sub-carrier is multiplied with the received signal, without the necessity to filter one of the sidebands. This is possible due to the quasi-orthogonality properties of the E5 primary codes.

When this type of method is used, the channels’ individual magnitude of correlation values is a BPSK-like triangle. If a coherent combination of pilot-only or data-only channels is used, which suffers from bit transition issues, the correlation waveform has a shape similar to the AltBOC(15,10) correlation waveform. However, the combination of the E5a and E5b sideband channels results in a BPSK-like correlation form, similar to the one achieved from DSB acquisition [8].

4.1.4 Direct AltBOC method

In this method, the reference signal generator employs a LUT (section 2.2.3) to combine all four signal components. The individual sub-carriers do not need to be generated and combined with the spreading codes, as the LUT essentially maps the sub-carrier phase points.

Since the data bits are not previously known, in order to resolve the data bit ambiguity, it is necessary to generate four LUT outputs, due to the existence of two data channels (E5a-I and E5b-I) and then perform the correlation four times (one for each LUT output). After all the correlations are complete, the maximum correlation value is then used in the detection process.
4.2 Categorization of the Search Strategies

The main difference between acquisition algorithms is in how the grid (section 3.2) is searched. This search can be performed in a variety of ways, which are introduced next [13].

4.2.1 Serial Search

In a serial search method, the cells are searched sequentially and every single cell of the grid is searched individually. This is the most basic method and it is used as a benchmark in algorithm comparison [8] [22]. Since all cells are searched individually, typically this method presents the slowest acquisition times. In section 4.3.1 an example of this type of acquisition is discussed.

4.2.2 Parallel Search

Parallel search methods allow a simultaneous search of the cells relative to at least one dimension (code delay or frequency shift). These methods are more complex than the serial search ones are, but they can achieve a faster acquisition.

Parallel Frequency Search

In this method, all the frequency bins are tested at the same time, which leaves only the code delays to be searched. This type of algorithm is faster than the benchmark as the number of bins searched is reduced. Section 4.3.2 presents an example for this type of algorithm.

Parallel Code Search

This search method is similar to the previous one, with the difference being that the simultaneous search is performed on the code delays. This method is faster than the previous one as the number of code delay bins is much higher than the number of frequency bins. However, its complexity is also higher due to the high number of bins searched at the same time. Section 4.3.3 shows this type of method.

Double Parallel Search

This type of algorithm simultaneously searches all the frequency bins and all the code delay bins, which means that it determines the correlation values for the whole search grid in one iteration. The methods used to achieve this are the most complex and difficult to implement, and in section 4.3.4 an example is presented.
4.3 SSB/DSB Acquisition Algorithms

In this section, the most appropriate algorithms for SSB and DSB acquisition are presented. The methods shown are only exemplified for the SSB acquisition, as these two types of acquisition are in all ways similar. The only difference resides in the fact that the correlations need to be calculated twice (one for the E5a and another for the E5b sidebands) and these results to be non-coherently combined. Also, the examples provided are shown for the acquisition on the E5a sideband, but they are also valid for the E5b sideband, with the respective parameters.

4.3.1 Classical/Serial Acquisition

The classical acquisition is the most basic acquisition scheme there is and it is described in a variety of references, such as [3] and [22]. This algorithm can be classified as a serial search algorithm as the correlation output is calculated for each cell individually. The acquisition scheme is shown in Figure 4.1. The scheme is used for an acquisition in a single channel: E5a-Q in this case.

**Figure 4.1: Architecture of the Serial Acquisition Algorithm for a Single Channel.**

In the algorithm described in Figure 4.1, where \( T_s \) stands for the sampling interval, the correlation is performed the same way it was presented in section 3.1. At first, the code wipe-off is performed by multiplying the incoming signal by a local code replica. Then, the carrier wipe-off is done by multiplying the signal by two sinusoidal generated signals, which result in in-phase and quadrature signals independent of the carrier frequency and the code delay. The correlation results are then summed over a coherent integration time of 1ms (equal to one secondary code chip time). After the envelope detector, the results are then non-coherently combined over an interval of K times the primary code period, which leads to the correlation output.

After the correlation outputs are determined for every single cell of the search grid, the maximum correlation value is taken and compared to a threshold, which determines if the signal is or not present.

This scheme can be modified to be able to perform the acquisition in both channels of the sideband, simply by performing the operations above twice (one for each signal) and then perform a non-coherent combination. The resulting scheme is present in Figure 4.2.

The rest of the algorithms are described for the acquisition of a single channel, but they are also true for the non-coherent integration of multiple channels, by performing a similar operation to what is shown.
4.3.2 Parallel Acquisition in Frequency Domain

This algorithm, as the name suggests, performs a parallel search in the frequency domain and its scheme can be seen in Figure 4.3.

![Figure 4.2: Architecture of the Serial Acquisition Algorithm for both Channels of a Sideband.](image)

![Figure 4.3: Architecture of the Parallel Acquisition in Frequency Domain Algorithm.](image)

The first part of the method consists in a code wipe-off, which is done in the same way as the classical acquisition. However, it is in the next part that the major difference appears: the carrier wipe-off is not performed. Instead, it is replaced by a Fourier Transform. This transformation changes the time domain data into frequency domain data. When the local generated code is aligned with the input data, the squared output of the transformed data shows a distinct peak, located at the index that corresponds to the frequency of the IF plus the Doppler shift. The envelope detector and the detection stage are in all similar to the previous algorithm.

The resolution that this algorithm can achieve is related to the number of FFT points. This number of points is directly related to the sampling frequency and the coherent integration time. The frequency resolution is then equal to [3] [13]

$$\Delta f_d = \frac{f_s}{\text{number of samples}} = \frac{f_s}{T_{coh} \times f_s} = \frac{1}{T_{coh}}.$$  (4.1)
This means that, for a standard coherent integration time of 1ms, the algorithm frequency resolution is equal to 1kHz. A frequency estimation with this resolution is usually not good enough to be passed to the tracking stage, so longer coherent integration times are advisable.

It must be noted that only the first half of the FFT components contain useful information, as the second half contains the complex conjugates of the first half [12]. This is important, because the second half of the components can be discarded allowing a faster acquisition.

### 4.3.3 Parallel Acquisition in Time Domain

This method has the search parallelized in time domain. To achieve this, it uses some Fourier Transform properties to obtain a circular correlation.

Let two N length signals be \( x(n) \) and \( y(n) \), their discrete N-point Fourier Transform can be expressed as:

\[
X(k) = \sum_{n=0}^{N-1} x(n)e^{-j2\pi kn/N} \\
Y(k) = \sum_{n=0}^{N-1} y(-n)e^{j2\pi k(-n)/N}. 
\]  

(4.2)

The cross correlation, \( z(n) \), between the signals \( x(n) \) and \( y(n) \) can be expressed as:

\[
z(n) = \bar{x}(-n) * y(n) \\
= \sum_{m=0}^{N-1} \bar{x}(-m)y(n - m),
\]

(4.3)

where \( \bar{x} \) denotes the complex conjugate of \( x \).

By applying a discrete Fourier transform to Equation (4.3), the following expression is obtained:

\[
Z(k) = \sum_{n=0}^{N-1} \sum_{m=0}^{N-1} \bar{x}(-m)y(n - m)e^{-j2\pi kn/N} \\
= \sum_{m=0}^{N-1} \bar{x}(-m) \sum_{n=0}^{N-1} y(n - m)e^{-j2\pi k(n-m)/N}e^{-j2\pi km/N} \\
= \sum_{m=0}^{N-1} \bar{x}(-m)Y(k)e^{j2\pi k(-m)/N} \\
= \bar{X}(k)Y(k).
\]

(4.4)

By performing the inverse Fourier transform, the cross correlation can be obtained by the following expression:

\[
z(n) = IFFT[\bar{X}(k)Y(k)].
\]

(4.5)

The parallel acquisition in time domain algorithm takes advantage of this relationship to perform the code correlation and its scheme is presented in Figure 4.4.

The method performs the carrier wipe-off the same way it is done in the classical method, being
the difference in the code wipe-off. The code wipe-off is performed using the relationship expressed in Equation (4.5), as it is shown in Figure 4.4.

This solution can be considered as an alternative to the classical implementation, that produced the same results just performing the correlation by means of FFT’s [3], which makes the acquisition faster.

4.3.4 Double Block Zero Padding Transition Insensitive (DBZPTI)

The DBZPTI method can be classified as double parallel search, since it parallelizes the search not only in the time domain but also in the frequency domain. This algorithm proposed in [23] corresponds to an updated and improved version of the algorithm presented in [24]. The biggest improvement consists in the fact that the new method suffers negligible degradation in the presence of bit transitions.

The full description of the algorithm is presented in [23]. The method can be summarized in five steps:

1. **Pre-processing of the incoming signal.** The incoming signal is converted to baseband through a multiplication by a sinusoidal at IF. The local sinusoidal does not try to compensate the incoming Doppler frequency as it is the case for the other acquisition methods. Then, the samples are split into 2M blocks of equal length. Each block contains the same number of samples (N). The M value depends on the integration time and on the width of the Doppler frequency interval and can be expressed as $M = 2f_{\text{max}} \times T_{\text{coh}}$, where $f_{\text{max}}$ corresponds to the maximum foreseeable Doppler shift. After that, each two adjacent blocks of the received signal are combined to form a block of samples (thus the name “Double Block”). The process is exemplified in Figure 4.5.

2. **Local replica code.** M blocks of the local spreading code are generated. Each block is zero-padded, this means that samples of value 0 are appended to each block. So the local code blocks have 2N samples as the incoming signal blocks. The process is exemplified in Figure 4.6.

3. **Partial circular correlation.** Each 2N-sample block of incoming signal is circularly correlated with the corresponding first zero-padded code block of the local replica. Due to the zero padding of the local code blocks, only the first half of the resulting correlation function is preserved. The resulting matrix has dimensions of MxN and the N points represent a partial correlation of length $\frac{T_{\text{coh}}}{M}$ ms at N possible delays. The process is exemplified in Figure 4.7.
4. **Circular correlation for all delays.** The incoming signal blocks are shifted one block and circular correlated with the unchanged local code blocks. This step is performed \( M \) times, for all the incoming signal combination of two blocks that correspond to each code delay. This results in \( M \times N \) matrices, that after being concatenated result in a single matrix with \( M \times (N \times M) \) dimensions. The step is illustrated in Figure 4.8.
5. **Zero padding and FFT application.** The resulting matrix is zero-padded by a chosen value. According to [24], the best value corresponds to 4M. This zero padding is done to achieve a better resolution in terms of Doppler frequency estimation. Finally, an FFT is performed for each column, one by one, which is representative of every code delay. The resulting matrix consists on the search space with all correlation values already determined.

![Diagram of zero padding and FFT application.](image)

**Figure 4.9: Zero padding and FFT application.**

### 4.3.5 Exploiting Secondary Codes Properties to Increase Acquisition Performance

All the acquisition architectures presented earlier have their coherent integration time limited to 1ms, corresponding to the duration of one secondary code chip. In this section, a method [8] is presented that allows the use of coherent times equal to \( T_{coh} = N_c \cdot T_p \), where \( N_c \) is the number of primary code periods chosen to be part of the coherent integration. Other advantage of this method is that the secondary code acquisition is performed simultaneously to the primary code acquisition.

The coherent accumulation is performed by using the knowledge of the secondary code, which means that an integration is performed for all \( L_S \) secondary code delays. The coherent integration is continued for the desired duration and the decision statistic is found by taking the maximum out of the \( L_S \) correlation values, which corresponds to the best approximation of the correct secondary code.
phase.

In this work, this method is applied only to one of the pilot channels. The reason is that, because the secondary codes of these channels are larger and all secondary codes are synchronized\(^1\), this is sufficient to determine all secondary code phases. Even though the secondary codes do not all have the same size, the larger codes, corresponding to the pilot channels, are multiples of the smaller ones, corresponding to the data channels. This means that the data channels secondary code phases can be determined as:

\[
\tau_{S_{e_{n-1}}} = \text{mod}(\tau_{S_{e_{n-2}}}, 20\, ms) \\
\tau_{S_{e_{n-1}}} = \text{mod}(\tau_{S_{e_{n-2}}}, 4\, ms).
\]

(4.6)

Since the method is only applied to the pilot channels, theoretically, the integration can be extended to any desired length as long as the receiver dynamics does not alter the code phase delay [8]. However, in practice, it is verified that long integrations introduce high computational burdens, which lead to slow acquisition times.

This method is possible due to two properties: the first one is described in section 2.1.2, where it is shown that, similarly to primary codes, the secondary codes in the pilot channels also present good auto-correlation properties; the second one is that it is not necessary to determine all bits of the secondary code to discover its phase.

This second property is related to the concept of characteristic length (CL). The characteristic length of binary sequence consists in the minimum number of bits necessary to reconstruct the entire sequence. This concept was introduced in [25] for pseudo-random sequences generated by a LFSR and it was called linear span. Even though the secondary codes are memory codes not generated by LFSR’s, this concept can be extended, being the span referred as characteristic length.

In [8] it is described a procedure to determine the CL of a given secondary code. It is as follows:

1. Determination of the number of contiguous zeros or ones (whichever is maximum) in the sequence with length 100 (length of a secondary code assigned to a pilot channel). Let that number be called \(k\).

2. Construction of a matrix \(M\) with the partial sequences of length \(k\) as the rows, where each is shifted one bit with respect to the previous row. This matrix has dimensions 100x\(k\).

\[
M = \begin{bmatrix} c_1 & c_2 & \ldots & c_k \\
& c_1 & c_2 & \ldots & c_{k+1} \\
\vdots & \vdots & \ddots & \vdots \\
& & & & c_{100-k} & c_{100-k+1} & \ldots & c_{100} \\
& & & & \vdots & \vdots & \ddots & \vdots \\
& & & & c_{100} & c_1 & \ldots & c_{k-1} \end{bmatrix}
\]

\(^1\)The start of one pilot channel secondary code corresponds to the start of both of the data channel secondary code.
3. Search on the matrix $M$ for identical rows. This search can be made by computing the linear rank coefficient matrix, $X = \text{CORR}(M)$, of the rows of $M$ and examining whether any entry of the matrix $X$ is unity. If two or more rows are found to be identical, then $k$ is incremented by one and step 2 is repeated until there are no two identical rows in matrix $M$.

4. The smallest value of $k$ that satisfies the conditions specified in step 3 is the CL of the sequence.

In Figure 4.10, the CL of all the secondary code sequences used in the E5 signal with length 100 are presented.

![Figure 4.10: CL of the Galileo E5 Secondary Codes with length 100 [8].](image)

It can be seen from the previous figure that the CL's are much smaller than the sequence lengths. The biggest length corresponds to 18, while the average CL is equal to 11.7, which corresponds, approximately, to 1/9 of the sequence length.

Figure 4.11 shows the full proposed algorithm. Depending on the value of the CL, $N_c$ may be smaller or larger than CL. The algorithm handles both situations appropriately has shown in the decision to the end the iteration below. The approach can be divided in five steps:

![Figure 4.11: Architecture of the Proposed Algorithm.](image)

**Branch generation**

In this step, the branches that contain all possible secondary code hypotheses are generated. The branches are generated following an evolutionary tree as described in [26] and represented in Figure 4.12. Each millisecond of input data is multiplied either by the zero bit or the one bit (+1 and -1,
respectively), as dictated by the corresponding branch value. In each iteration, only the $i^{th}$ millisecond of input data is updated, where $i$ is the iteration number.

![Figure 4.12: Secondary Code Evolutionary Tree at the Third Iteration [26].](image)

Following each iteration, each branch of the tree is divided in two: in one a zero is evaluated and in the other a one is evaluated. Since the output of the correlator is squared, there is no distinction between a negative and positive value, which means that only one half of the tree needs to be evaluated. The other half contains the same information as the first one.

This means that, in each iteration, the number of branches that are created and in need to be evaluated are:

$$B_i = 2^{i-1} - E_i,$$

where $E_i$ is the number of branches eliminated in the iteration $i$. Obviously, in the first iteration $E_1 = 0$.

**Primary Code Correlation**

The primary code correlator performs the correlation of the input signal with local primary code replica. In this step, any of the previously discussed acquisition methods can be used. In each iteration, the correlation is performed over the first $i$ milliseconds of data for each one of the branches present in the evolutionary tree. Only the maximum output for each branch (and its corresponding position in the search space) is saved during this step.

**Branch Elimination**

The branch elimination logic examines all the hypotheses outputs, corresponding to each one of the branches. The criterion for the branch elimination is the lower correlation value relative to other branches. Let $S = (\hat{s}_1, \hat{s}_2, \ldots, \hat{s}_{B_i})$ be the vector containing the correlation values for each one the branches, where $\hat{s}_n$ is the correlation value of the branch $n$. A vector $D$ containing the difference between $S$ and its maximum is created:

$$D = \max(S) - S.$$

At each iteration, $E_i$ branches whose $D$ value exceeds a predefined threshold are eliminated. After a branch is eliminated, all its successors are also considered eliminated.
Decision to end the iteration

This step can be split in two cases:

- $N_c < CL$. In this case, the primary code and corresponding code shift are detected before the secondary code phase is determined. Here, there are two choices: either the iterations are continued until, at least, the corresponding CL number of secondary bits are determined or the process stops and the secondary code shift is acquired in a different way (see section 4.6). Usually, the first choice is made.

- $N_c > CL$. In this case, the integration is performed either until a correlation value exceeds the decision threshold (signal present) or until the iteration number exceeds $N_c$ (signal not present).

Secondary Code Chip Position Retrieval

The secondary code hypotheses that wins (maximum correlation value at the end of the iterations) corresponds to a sub-sequence within the complete secondary code. A search needs to be made in the larger sequence to determine the index of the shift. Note that the search needs to be made not only with the winner hypothesis but also with its symmetrical due to the way that the correlation is performed.

4.4 FIC Acquisition Algorithm

In this section, a full-band independent code acquisition algorithm is presented. This method is called Sub-carrier Phase Cancellation (SPC), which is proposed and fully described in [27].

4.4.1 Sub-carrier Phase Cancellation Method

The idea behind the SPC technique is to get rid of the sub-carrier signal, the same way it is done for the carrier component. To achieve this, in addition to the local in-phase and quadrature signals used to perform the carrier wipe-off, it is necessary to generate an in-phase and a quadrature sub-carrier signals. The expression for the local in-phase sub-carrier signal, $r_I$, and quadrature sub-carrier signal, $r_Q$, are as follows [27]:

\[
\begin{align*}
  r_I &= c(t - \hat{\tau}).sc(t - \hat{\tau}) \\
  r_Q &= c(t - \hat{\tau}).sc(t - \hat{\tau} - \frac{T_{sc}}{4}),
\end{align*}
\]

where $c(t)$ can correspond to the primary code of any of the four channels.

The method architecture is shown in Figure 4.13. The algorithm represented performs the acquisition of a single channel (E5a pilot), but it can be extended to all the cases described in section 4.1.3.

It turns out that when the correlation outputs of the two channels, $S_I$ and $S_Q$, are combined to form a reconstructed ACF, the shape of the correlation function is similar to the BPSK triangle. This is shown in Figure 4.14.
4.5 Direct AltBOC Acquisition Algorithms

The acquisition on the full E5 band is performed in a similar way to the acquisition of a single band. However, there are some major differences. First of all, as shown in section 2.3.4, the sampling rates used need to be significantly higher, even though it makes possible to achieve better time resolution, it leads to higher computational burden, and therefore, slower acquisition times.

In order to decrease the high acquisition time, it is proposed in [9] the progressive acquisition tech-
nique. Here, the input signal is down-sampled to a lower sampling rate (about 2 samples/chip [9]), so that the acquisition technique obtains an initial rough estimate of the code delay and Doppler shift. After that, the acquisition is performed for the regular sampled signal, but only around the cell that achieved the largest correlation value, increasing the resolution to the appropriate level.

As stated in section 3.1, the code wipe-off needs to be performed in a different way. Instead of being performed for a single primary code, it needs to executed for the full modulation. To do this, the local replica is generated using the LUT approach, discussed in section 2.2.3, as to combine all four signal components. It is not necessary to generate the individual sub-carriers to combine with the individual channels codes, as the LUT essentially maps the sub-carrier phase points [8].

The LUT approach suffers from data bit ambiguity. One method to solve this issue, is to form four LUT outputs, that correspond to the for four possible data bit combinations, due to the two data channels (E5a-I and E5b-I). The correlation is performed for the four LUT outputs and the maximum correlation value, corresponding to the correct data bit combination, is selected to be part of the detection stage.

As stated before, due to the secondary codes, the coherent integration limit of the acquisition can not exceed 1ms. Although, in section 4.3.5, it was discussed a method to increase this limit, that particular method can not be applied for the direct acquisition. Theoretically, there is nothing that impedes the correct functioning of the algorithm, however, since the local signal is the result of the combination of four different codes, the number of branches in the evolutionary tree is equal to $16^i$ (vs $2^{i-1}$ for a single component). Now, the deployment of this algorithm for this number of branches leads to excessive acquisition times that are impractical for the process.

So, [9] proposes to perform the method discussed in section 4.3.5 in two steps:

1. Acquire one of the pilot components with a coherent integration time sufficient to acquire the secondary code shift position.

2. Use the information of the secondary chip position and perform the direct acquisition in the full E5 band.

### 4.6 Transition to Tracking

During the acquisition, the receiver estimates both the code delay and the frequency shift of the received signal. The next stage of the receiver (tracking) requires a lock into a code loop, called Delay Locked Loop (DLL), and a carrier loop, called Phase Locked Loop (PLL) to correctly track the signal from a given satellite. A transition stage is required between these two stages because of two issues:

- The acquisition is performed only for the primary codes, but the secondary code phase must also be known.

- The estimated parameters may not have the required resolution.

If the algorithm used in section 4.3.5, where the secondary code is acquired simultaneously to the primary code, is not used, the secondary code phase acquisition is performed after the primary code
is already acquired. As stated in section 4.3.5, the secondary code acquisition only needs to be done for a single pilot channel, as all other secondary code phases can be directly obtained from it. After the primary code is acquired, the secondary code chips can be demodulated, by checking the signal of a correlation value in a given secondary code chip, and stored in a vector. The secondary code synchronization can then be obtained, performing a circular correlation between the vector with the estimated chips and a local replica of the secondary code [9].

The second issue that must be solved is due to the resolution being too rough, due to the need to reduce the computational burden. Conventional GNSS receivers solve this problem by using a convergence phase, where the carrier loop is a Frequency Locked Loop (FLL), to refine the frequency estimation and to allow the successive phase lock (PLL) and the DLL to work in a coarse configuration to tolerate the initial code phase errors. This is also true for the E5 AltBOC modulation if a SSB or DSB acquisition is performed.

Due to the multi-peak auto-correlation function of the AltBOC modulation, which can lead to a false lock on a secondary peak, when the direct acquisition is performed an additional care needs to be taken. In this case, the correlation function can be directly applied, using a bump-jumping technique to solve the ambiguity problem. An example of this technique is presented in [27] and consists on using multiple correlators with early, prompt and late versions of the codes, in order to estimate the peaks power and, consequently, choose the strongest peak, which is the main one.
Chapter 5

Performance Analysis

In this chapter, the performance of algorithms discussed in the previous chapter is analyzed. The analysis focuses on the time the acquisition takes to be complete and the acquisition sensibility, i.e. statistical analysis of the algorithms are made. At the end of the chapter all the results are summarized and compared (section 5.6) and an analysis is made on the effect of the integration time (section 5.7).

5.1 Signal generation

The generated signals are produced using a script implemented in the MATLAB language. These signals are generated as if they are the output of the Analog-to-Digital Converter.

The modulated signals follow Equation (2.17) and are generated for a chosen code delay. Let $s_{E5}(t) = R(t) + jI(t)$ stand for the AltBOC(15,10) baseband complex signal with $R(t)$ and $I(t)$ representing, respectively, the real and imaginary parts. The receiver RF signal affected by the Doppler frequency shift $\omega_d$ is

$$r(t) = A[R(t) \cos((\omega_0 + \omega_d)t + \phi) - I(t) \sin((\omega_0 + \omega_d)t + \phi)] + w(t),$$

(5.1)

where $A$ is the amplitude, $\omega_0$ is the nominal carrier frequency, $\phi$ is an additional phase, and $w(t)$ is additive white Gaussian noise with power spectral density $G_w(f) = N_0/2$. The corresponding IF signal is

$$r_{IF}(t) = A[R(t) \cos((\omega_{IF} + \omega_d)t + \phi) - I(t) \sin((\omega_{IF} + \omega_d)t + \phi)] + \tilde{w}(t),$$

(5.2)

where $\omega_{IF} = 2\pi f_{IF}$ is the intermediate frequency.

The wideband Gaussian noise, shifted to $f_{IF}$, is given by

$$\tilde{w}(t) = n_i(t) \cos(\omega_{IF}t + \phi) - n_q(t) \sin(\omega_{IF}t + \phi),$$

(5.3)

where $n_i(t)$ and $n_q(t)$ are, respectively, the in-phase and quadrature components of the input noise $w(t)$ with power spectral density equal to $N_0$. If $B$ is the bandwidth of the bandpass signal $r_{IF}$, then the mean
The power of \( \tilde{w}(t) \) is
\[
P_{\tilde{w}} = E\{\tilde{w}^2(t)\} = \frac{1}{2}[E\{n_i^2(t)\} + E\{n_q^2(t)\}] = N_0B.
\] (5.4)

The signal-to-noise ratio of \( r_{IF}(t) \) is equal to
\[
SNR = \frac{P_S}{P_{\tilde{w}}},
\] (5.5)

where \( P_S \) is the power of the AltBOC signal in \( r(t) \).

Thus, \( C/N_0 = P_S/N_0 \) and
\[
SNR = \frac{1}{B} \left( \frac{C}{N_0} \right)
\] (5.6)
is the signal-to-noise ratio at the output of the IF stage.

The input filters used in the single band receiver and separate double band receiver are considered ideal, which means that the unwanted sideband signals are completely eliminated, while the wanted signals suffer no attenuation.

A frequency shift is introduced on the signal equal to \( f_{IF} + f_D \), simulating the effects of both the intermediate and Doppler frequencies. Both values are chosen prior to the signal generation, according to the discussion made in section 2.3.4. The chosen values for the intermediate frequency were 15 MHz for the single band receiver and 30 MHz for the coherent double band receiver.

Then, the signal is sampled at a chosen sampling frequency, appropriate to the acquisition. The simulated outputs of the ADC have float precision, which means that the ADC is considered to have an infinite number of bits. The effects of using an ADC with a finite number of bits is not studied in this work, but a study on the subject was made in [3].

Finally, an Additive White Gaussian Noise (AWGN) channel is simulated, by adding, to the signal samples, a Gaussian noise with zero mean and variance defined by the chosen SNR, or the corresponding \( C/N_0 \) at the input of the acquisition step.

These steps are performed for every satellite chosen to be visible and the resulting signals are added to generate the input signal which is stored in a text file.

### 5.1.1 Carrier-to-Noise Density Ratio

In order to obtain results that are as similar as possible as those obtained using real signals, a proper choice of the \( C/N_0 \) must be made. According to [5], the minimum received power on ground of the Galileo signals measured at the output of an ideally matched RHCP 0 dBi polarized user receiving antenna when the satellite elevation angle is higher than 10 degrees is \( P_{\text{sig}} = -155dBW \) for the E5 band. Knowing that the noise power of the received signal, \( N \), is equal to \( N = N_0 \times B_{\text{inp}} \), where \( N_0 \) corresponds to the power spectral density of the input noise and \( B_{\text{inp}} \) to the receiver input bandwidth, and that, according to [10], the \( N_0 \) at a typical GNSS receiver is equal to \( 7.81 \times 10^{-21} \) W/Hz, it can be assumed that the noise power is equal to
\[
N = 7.81 \times 10^{-21} \times B_{\text{inp}}.
\] (5.7)
As seen in chapter 2.3, the input bandwidth is dependent on the type of receiver considered. The input bandwidth can either be 20 MHz or 51 MHz so, for those input bandwidths, the noise power is equal to

\[ N_{sb} = 7.81 \times 10^{-21} \times 20 \times 10^6 = 1.56 \times 10^{-13} \text{ W} = -128 \text{ dBW} \]

\[ N_{cdb} = 7.81 \times 10^{-21} \times 51 \times 10^6 = 3.98 \times 10^{-13} \text{ W} = -124 \text{ dBW}, \]

where the index \( sb \) is related to a single band receiver and the index \( cdb \) to the coherent double band receiver.

The signal to noise ratio is given by the ratio between the signal power and noise power, as follows:

\[ S/N_{sb} = \frac{P_{sig}}{N_{sb}} = -155 \text{ dBW} - 128 \text{ dBW} = -27 \text{ dB} = 2.0 \times 10^{-3} \]

\[ S/N_{cdb} = \frac{P_{sig}}{N_{sb}} = -155 \text{ dBW} - 124 \text{ dBW} = -31 \text{ dB} = 7.9 \times 10^{-4}. \]

Note that these values are very low because the GNSS signal was not despreaded. This occurs only after the multiplication by the code sequence.

The C/N\(_0\) can be taken from the signal to noise ratio, simply by multiplying it by the input bandwidth, so:

\[ C/N_{0, sb} = S/N_{sb} \times B_{sb} = 2.0 \times 10^{-3} \times 20 \times 10^6 = 4.0 \times 10^4 = 46 \text{ dB.Hz} \]

\[ C/N_{0, cdb} = S/N_{cdb} \times B_{cdb} = 7.9 \times 10^{-4} \times 51 \times 10^6 = 4.0 \times 10^4 = 46 \text{ dB.Hz}. \]

The obtained C/N\(_0\) values are used as the reference in the study of the performance of the algorithms.

### 5.2 Classical Acquisition Algorithm

The classical acquisition was implemented using the schematic presented in Figure 4.1. The obtained results are summarized next.

#### 5.2.1 Simulation Parameters

The classical acquisition algorithm was implemented using the parameters described in Table 5.1.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Integration Time</td>
<td>1 ms</td>
</tr>
<tr>
<td>Doppler range</td>
<td>±5 kHz</td>
</tr>
<tr>
<td>Primary code length</td>
<td>10230</td>
</tr>
<tr>
<td>Doppler step</td>
<td>500 Hz</td>
</tr>
<tr>
<td>Code step (ssb/dsb acquisition)</td>
<td>0.250 T(_c)</td>
</tr>
<tr>
<td>Code step (direct acquisition)</td>
<td>0.07692 T(_c)</td>
</tr>
<tr>
<td>Sampling rate (ssb/dsb acquisition)</td>
<td>4/T(_c)</td>
</tr>
<tr>
<td>Sampling rate (direct acquisition)</td>
<td>13/T(_c)</td>
</tr>
</tbody>
</table>

Table 5.1: Parameters used in the classic acquisition.
These parameters were chosen according to what was discussed in the previous chapters, more concretely in chapter 2 and chapter 3. The direct acquisition requires a smaller code step, due to the AltBOC ACF whose main peak is narrower than the peak of the BPSK modulation.

5.2.2 Mean Acquisition Time

The mean acquisition time (MAT) is related to the number of bins in the search space that need to be looked at and to the time it takes to analyze each one of the cells. So, the acquisition time is equal to:

\[
MAT = \text{number of search cells} \times \text{time needed to search 1 cell}. \quad (5.11)
\]

The number of search cells is given by

\[
\text{number of search cells} = \text{Doppler bins} \times \text{code bins} = \frac{\text{Doppler range}}{\text{Doppler step}} \times \frac{\text{length of the primary code}}{\text{code step}} \quad (5.12)
\]

Using the parameters of Table 5.1, the total number of cells is equal to

\[
\begin{align*}
\text{number of search cells (ssb/dsb)} &= \left[10^4\frac{500}{500} + 1\right] \times \frac{10230}{0.250} = 859320 \text{ cells} \\
\text{number of search cells (direct)} &= \left[10^4\frac{500}{500} + 1\right] \times \frac{10230}{0.07692} = 2792790 \text{ cells}. \quad (5.13)
\end{align*}
\]

As seen in Equation (5.13), the number of cells searched in the direct acquisition is really high, when compared to the SSB acquisition. So to decrease this number, the progressive acquisition method (section 4.5) is implemented. Using a code step of 0.25 for the first iteration and the usual code step for the second and last iteration\(^1\), the total number of cells is

\[
\text{number of search cells (direct)} = \left[10^4\frac{500}{500} + 1\right] \times \frac{10230}{0.25} + 3 \times \frac{2}{0.07692} = 859398 \text{ cells}, \quad (5.14)
\]

which represents a reduction of about 70% on the number of cells.

The time needed to perform the correlation for a search cell, \(T_{\text{cell}}\), not only depends on the algorithm but also on the hardware that processes the data. So, to eliminate the influence of the hardware, the time results (not only in this method but also for the rest of the methods) are normalized to a reference. The reference chosen is the time it takes to perform the correlation on one cell using the classical acquisition algorithm for a single pilot channel.

The time results were obtained by averaging 1000 different measurements and are presented in Table 5.2. The MAT is calculated using Equation (5.11) with the number of cells normalized by the number of cells used in the classical acquisition of a single pilot channel.

From Table 5.2, it can be seen that there is a massive increase in the acquisition time when the direct acquisition is performed. This increase is not only due to the increase in the number of signal channels

\(^1\)The last iteration performs the search around the maximum of the correlation value obtained from the first iteration. The last search is made in a radius of one primary code chip and one Doppler step.
Table 5.2: Time results for the classic acquisition.

acquired but also due to the increased sampling rates used, to the need to map the modulation using the LUT implementation and to the need to repeat the correlation process four times, so that the influence of the navigation bits is neglected.

5.2.3 Simulation Results

To evaluate the performance of the algorithm, signals were generated with a $C/N_0$ value of 46dB.Hz (section 5.1.1). The result of the SSB algorithm for a single pilot channel is presented in Figure 5.1. The data were generated with a Doppler shift of 1kHz and a code shift of $0.25 \times T_c$.

As it can be seen in Figure 5.1, in regular conditions, an integration time of 1ms of data is not enough to distinguish the received signal data from the noise. To solve this issue, the results of ten primary code periods were non-coherently combined ($K' = 10$)$^2$. The new results are shown in Figure 5.2. To make a fair comparison between the algorithms, the following simulations are also performed for a non-coherent integration time of 10ms.

In Figure 5.2, there is, now, a clear correlation peak (corresponding to the correct combination of parameters) in the search grid. Figure 5.2(b) shows a zoomed cut on the time domain of the correlation peak of Figure 5.2(a). It is seen that the correlation peak has an approximate shape of a triangle, which is characteristic of the BPSK modulation, as expected.

$^2$In section 5.7, a study on the integration time is performed.
To better understand the results of the algorithm, the simulation was performed 1000 times both for a present signal and for an absent signal and, in Figure 5.3, an histogram with the maximum correlation values of both cases is presented. The threshold was determined using the method described in section 3.3, for a probability of false alarm: $P_{fa} = 10^{-2}$.

From Figure 5.3, it can be seen that there is a considerable difference between the correlation values when there is and when there is not a signal present, which is good for the detection stage. For this particular case, the probability of detection is close to one, as all the correlation values obtained when there is a signal present are higher than the set threshold. The same study was performed for the DSB acquisition of both pilot channels and for the direct correlation of the AltBOC modulation. The results are shown in Figure 5.4.

From Figure 5.4(a) it can be observed that there is an increase on both the components of the present and absent signals when compared to the SSB acquisition (see Figure 5.3). This is an expected result, since the DSB acquisition implemented is, basically, a non-coherent integration of two pilot channels.
Figure 5.4: Histograms of the classical acquisition results at $C/N_0 = 46$dB.Hz.

Figure 5.4(b) presents the results for the direct acquisition. Now, this is the most promising type of acquisition in terms of sensibility, since the correlation values caused by noise (signal not present) are at the same level of the SSB acquisition, but the signal components have much higher correlation values than for the rest of the cases.

A similar study was done for a variety of $C/N_0$ and a relation between the probability of correct detection and the $C/N_0$ of the received signal was obtained, which can be observed in Figure 5.5. The probability of detection was determined by checking how many of the correlation values of the known to be present signal exceeded the set threshold. Each data point represented in Figure 5.5 is the result of 1000 runs of the algorithm.

Figure 5.5: Sensitivity of the Classical Acquisition.

Figure 5.5 compares the algorithm sensitivity to the variation of the $C/N_0$. It can be observed that the DSB acquisition presents a slightly better performance, as expected, than the SSB acquisition. However, the highlight is on the direct acquisition: it presents exceptional results even in high noise conditions where the other two examples can not operate.
A particularity about the direct acquisition is the possibility to use a progressive acquisition method to analyze less cells and therefore accelerate the acquisition times. This method is exemplified in Figure 5.6.

Figure 5.6: Example of a direct acquisition performed with a progressive acquisition technique.

As previously explained, the progressive acquisition performs a first rough estimation to detect an approximate position of the correlation peak. After the peak is found, a more thorough search is made, but only around the detected peak, such as presented in Figure 5.6(b). Similarly to what was done for the SSB acquisition, a zoomed cut on the time domain of the peak can be made. This cut is shown in Figure 5.7.

Figure 5.7: Zoomed Normalized Correlation of the Direct Acquisition Peak.

The correlation function represented in Figure 5.7 resembles the theoretical auto-correlation function of an AltBOC signal with some distortion caused by the noise and sampling process. It presents the most relevant attributes of the modulation, which are the existence of multi-peaks and the narrow primary code peak.
5.3 Parallel Search in Frequency Domain Algorithm

The parallel search in frequency domain acquisition was implemented using the schematic presented in figure 4.3. The obtained results are summarized below.

5.3.1 Simulation Parameters

The algorithm was implemented using the parameters shown in Table 5.3.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Integration Time</td>
<td>1 ms</td>
</tr>
<tr>
<td>Primary code length</td>
<td>10230</td>
</tr>
<tr>
<td>Code step (ssb/dsb acquisition)</td>
<td>$0.250 \ T_c$</td>
</tr>
<tr>
<td>Code step (direct acquisition)</td>
<td>$0.07692 \ T_c$</td>
</tr>
<tr>
<td>Sampling rate (ssb/dsb acquisition)</td>
<td>$4/T_c$</td>
</tr>
<tr>
<td>Sampling rate (direct acquisition)</td>
<td>$13/T_c$</td>
</tr>
</tbody>
</table>

Table 5.3: Parameters used in the parallel search in frequency domain.

Since the integration time $T_{int}$ used has duration of 1 ms, the algorithm frequency resolution, according to Equation (4.1), is equal to 1 kHz. The maximum detectable frequency corresponds to half the sampling frequency.

5.3.2 Mean Acquisition Time

As stated in the previous section, the MAT depends on the number of searched cells. In this algorithm, the number of cells searched is smaller than for the classical acquisition, as the search is not performed in the frequency domain, this is, in each iteration, the correlation values are calculated for every frequency value. This leads to a decrease in the number of search cells which is equal to

$$\text{number of search cells (ssb/dsb)} = \frac{10230}{0.250} = 40920 \text{ cells}$$

$$\text{number of search cells (direct)} = \frac{10230}{0.0250} + \frac{3}{0.07692} = 40959 \text{ cells},$$

where the direct acquisition also includes the progressive acquisition method.

Including the $T_{cell}$, the time results can be summarized in Table 5.4.

<table>
<thead>
<tr>
<th>Acquisition Type</th>
<th>Channels</th>
<th>$T_{cell}$</th>
<th>MAT</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSB</td>
<td>pilot</td>
<td>0.609</td>
<td>0.0290</td>
</tr>
<tr>
<td></td>
<td>data + pilot</td>
<td>0.797</td>
<td>0.0379</td>
</tr>
<tr>
<td>DSB</td>
<td>pilots</td>
<td>0.925</td>
<td>0.0440</td>
</tr>
<tr>
<td></td>
<td>data + pilots</td>
<td>1.080</td>
<td>0.0505</td>
</tr>
<tr>
<td>Direct</td>
<td>all 4</td>
<td>91.233</td>
<td>4.3518</td>
</tr>
</tbody>
</table>

Table 5.4: Time results for the parallel search in frequency domain acquisition.

The results presented show that this method, because it performs less iterations (smaller number of cells searched), is computationally faster and demonstrates superior performance than the classical method.
5.3.3 Simulation Results

The performance of the algorithm was evaluated by applying the method to generated signals with \( \frac{C}{N_0} \) equal to 46 dB.Hz. However, an issue appeared during the execution of the algorithm. Due to the high sampling frequencies involved, there was the need to store the correlation results in matrices with dimensions too high (for example, \( 40920 \times 20460 \) in the SSB case) to be supported by MATLAB. The solution found was to divide the search space in “blocks”, dividing the time domain. This way, during its execution, the algorithm only stores the correlation values from the current block while keeping only the maximum correlation value from the previous “blocks”. In Figure 5.8, it is presented the output of the algorithm for the block which holds the maximum correlation value.

![SSB Algorithm Output](image)

**Figure 5.8:** SSB Algorithm Output.

It must be noted that the search made in the frequency domain of the algorithm does not find the Doppler shift directly. Instead, since the input signal is not converted to baseband, the maximum correlation value occurs for a frequency equal to the sum of the intermediate frequency with the Doppler shift.

Similarly to what was done in the analysis of the classical acquisition, the histograms which hold the maximum correlation values obtained during 1000 repetitions of the algorithm, using signals with similar noise floors, are presented in Figure 5.9.

These results are similar to the classical algorithm, since there is a clear distinction between the values when the signal is and when it is not present. A study on the probability of detection was also performed and its results are presented in Figure 5.10, where each data point is the result of 1000 runs of the algorithm.
Figure 5.9: Histograms of the parallel search in frequency domain acquisition results at $C/N_0 = 46\text{dB.Hz}$.

Figure 5.10: Sensitivity of the Parallel Search in Frequency Domain Algorithm.
5.4 Parallel Search in Time Domain Algorithm

The implementation of this method uses the algorithm schematic described in Figure 4.4. The algorithm performance is studied below.

5.4.1 Simulation Parameters

The algorithm was implemented using the parameters shown in Table 5.5.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Integration Time</td>
<td>1 ms</td>
</tr>
<tr>
<td>Doppler range</td>
<td>±5 kHz</td>
</tr>
<tr>
<td>Primary code length</td>
<td>10230</td>
</tr>
<tr>
<td>Doppler step</td>
<td>500 Hz</td>
</tr>
<tr>
<td>Sampling rate (ssb/dsb acquisition)</td>
<td>$4/T_c$</td>
</tr>
<tr>
<td>Sampling rate (direct acquisition)</td>
<td>$13/T_c$</td>
</tr>
</tbody>
</table>

Table 5.5: Parameters used in the parallel search in time domain.

In this algorithm, the code step is related to the sampling frequency and the code resolution that can be achieved is equal to $\frac{1}{R_s}$, where $R_s$ is the sampling rate.

5.4.2 Mean Acquisition Time

In this method, the search is not performed in the time delay, being the frequency domain the only one to be searched. Since the number of code delay bins is typically much higher than the number of frequency bins in the classical acquisition, this method presents a high reduction in the number of searched cells, which is equal to

$$\text{number of search cells} = \frac{10^4}{500} + 1 = 21 \text{ cells.} \quad (5.16)$$

Including the $T_{\text{cell}}$ which was calculated from 1000 different runs of the algorithm, the time results can be summarized in Table 5.6.

<table>
<thead>
<tr>
<th>Acquisition Type</th>
<th>Channels</th>
<th>$T_{\text{cell}}$</th>
<th>MAT</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSB</td>
<td>pilot</td>
<td>2.218</td>
<td>$5.420 \times 10^{-6}$</td>
</tr>
<tr>
<td></td>
<td>data + pilot</td>
<td>2.721</td>
<td>$6.650 \times 10^{-5}$</td>
</tr>
<tr>
<td>DSB</td>
<td>pilots</td>
<td>4.127</td>
<td>$1.008 \times 10^{-4}$</td>
</tr>
<tr>
<td></td>
<td>data + pilots</td>
<td>5.038</td>
<td>$1.231 \times 10^{-4}$</td>
</tr>
<tr>
<td>Direct</td>
<td>all 4</td>
<td>119.820</td>
<td>$2.928 \times 10^{-3}$</td>
</tr>
</tbody>
</table>

Table 5.6: Time results for the parallel search in time domain acquisition.

Even though that $T_{\text{cell}}$ is higher than the other algorithms, since the number of cells is significantly lower, the time results end up being much better. This can be justified based on the fact that the number of code bins is much bigger than the number of Doppler bins. Since the code bin search is eliminated in this method, the acquisition time becomes faster.
5.4.3 Simulation Results

Similarly to what was done for the previous algorithms, at first it was studied the algorithm’s performance for a \( C/N_0 \) of 46dB.Hz. The output of one simulation of the algorithm is shown in Figure 5.11.

![Figure 5.11: SSB Algorithm Output.](image)

The statistical analysis of 1000 runs of the algorithm using signals with the same \( C/N_0 \) is represented in the histograms of Figure 5.12.

![Histograms of the parallel search in time domain acquisition results at \( C/N_0 = 46 \text{ dB.Hz.} \)](image)

Figure 5.12: Histograms of the parallel search in time domain acquisition results at \( C/N_0 = 46 \text{ dB.Hz.} \)
The results of this algorithm are practically identical to the ones of the classical acquisition. This is due to fact that, as explained in section 4.3.3, the operations being performed on both algorithms are mathematically equivalent. The algorithm sensitivity study is shown in Figure 5.13, and as expected, the results are identical to the classical case.

![Figure 5.13: Sensitivity of the Parallel Search in Time Domain Algorithm.](image)

5.5 Double Block Zero Padding Transition Invariant Algorithm

The DBZPTI method is implemented such as described in section 4.3.4. The algorithm performance is studied below.

5.5.1 Simulation Parameters

The DBZPTI acquisition algorithm was implemented using the parameters described in Table 5.7.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Integration Time</td>
<td>1 ms</td>
</tr>
<tr>
<td>Doppler range</td>
<td>±5 kHz</td>
</tr>
<tr>
<td>Primary code length</td>
<td>10230</td>
</tr>
<tr>
<td>Matrix zero-pad</td>
<td>4M</td>
</tr>
<tr>
<td>Sampling rate (ssb/dsb acquisition)</td>
<td>4/T_c</td>
</tr>
<tr>
<td>Sampling rate (direct acquisition)</td>
<td>13/T_c</td>
</tr>
</tbody>
</table>

Table 5.7: Parameters used in the DBZPTI algorithm.

The resulting code step is equal to the one on the parallel search in the time domain which is the inverse of the sampling rate. For a matrix zero-pad value of 4M, the obtained frequency step is equal to 200 Hz.
5.5.2 Mean Acquisition Time

Since the algorithm performs a double parallel search, there is only one searchable cell, which includes all Doppler and code bins. This means that the algorithm only does one iteration. The time results can be summarized in Table 5.8.

<table>
<thead>
<tr>
<th>Acquisition Type</th>
<th>Channels</th>
<th>( T_{cell} )</th>
<th>MAT</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSB</td>
<td>pilot</td>
<td>29.962</td>
<td>( 3.486 \times 10^{-5} )</td>
</tr>
<tr>
<td></td>
<td>data + pilot</td>
<td>53.970</td>
<td>( 6.280 \times 10^{-5} )</td>
</tr>
<tr>
<td>DSB</td>
<td>pilots</td>
<td>84.759</td>
<td>( 9.863 \times 10^{-5} )</td>
</tr>
<tr>
<td></td>
<td>data + pilots</td>
<td>130.391</td>
<td>( 1.517 \times 10^{-4} )</td>
</tr>
<tr>
<td>Direct</td>
<td>all 4</td>
<td>1 239.586</td>
<td>( 1.442 \times 10^{-3} )</td>
</tr>
</tbody>
</table>

Table 5.8: Time results for the DBZPTI acquisition.

Compared to the previous time results, it can be verified that \( T_{cell} \) has the slowest value in this algorithm. However, since only one cell is analyzed during the whole process, it ends up that the algorithm achieves the fastest MAT.

5.5.3 Simulation Results

The algorithm was, at first, tested for a single pilot channel at the reference \( C/N_0 \). The resulting search space is presented in Figure 5.14.

![Figure 5.14: SSB Algorithm Output.](image)

The same tests were performed as for the other algorithms and its results are presented below. Figure 5.15 shows the statistical histograms, while Figure 5.16 exhibits the algorithm sensitivity.
Figure 5.15: Histograms of the DPZPTI algorithm at $C/N_0 = 46$ dB.Hz.

Figure 5.16: Sensitivity of the DBZPTI Algorithm.
5.6 Algorithm Comparison

The time results obtained in the previous sections are summarized and presented in Table 5.9.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>SSB</th>
<th>DSB</th>
<th>Direct</th>
</tr>
</thead>
<tbody>
<tr>
<td>Classical</td>
<td>1</td>
<td>2.004</td>
<td>63.692</td>
</tr>
<tr>
<td>Parallel search in frequency</td>
<td>0.0290</td>
<td>0.0440</td>
<td>4.3518</td>
</tr>
<tr>
<td>Parallel search in time</td>
<td>5.420 × 10⁻⁵</td>
<td>1.008 × 10⁻⁴</td>
<td>2.928 × 10⁻³</td>
</tr>
<tr>
<td>DBZPTI</td>
<td>3.486 × 10⁻⁵</td>
<td>9.863 × 10⁻⁵</td>
<td>1.442 × 10⁻³</td>
</tr>
</tbody>
</table>

Table 5.9: Normalized MAT for all the studied algorithms.

From Table 5.9, it can be seen that the DBZPTI algorithm is the method that achieves the fastest acquisition times, having some advantage over the parallel search in time domain, while the classical method performs the slowest acquisition. Also, there is a considerable increase in the acquisition times when the direct acquisition is performed, independently of the algorithm considered.

Figure 5.17 shows the sensitivity comparison of the algorithms considered for the SSB (Figure 5.17(a)), DSB (Figure 5.17(b)) and direct (Figure 5.17(c)) acquisitions.

As it can be seen from Figure 5.17, there is little difference between the sensitivity of the algorithms.
The major difference lies on how and which signals are acquired, where the direct acquisition clearly outperforms both the SSB and DSB acquisitions.

5.7 Analysis on the Integration Time

In this section, a study on the effect of the integration time in acquisition performance is made. The study is performed solely for the acquisition of a single pilot channel using the DBZPTI method, since this method proved to be the fastest while presenting similar sensitivity to the other methods.

Figure 5.18 shows the detection probability versus the C/N₀ for one acquisition channel using a variety of non-coherent integration times (K primary code periods) with \( T_{int} = 1 \) ms (Figure 5.18(a)) and a variety of \( T_{int} \) for \( K = 1 \) (Figure 5.18(b)).

From the analysis of Figure 5.18, it can be seen that, as expected, the acquisition achieves better results when the coherent integration is performed instead of non-coherent one. However, unlike the implementation of the non-coherent combination which can be done directly, to be able to use coherent integration times superior to 1 ms, it was necessary to implement to algorithm described in section 4.3.5.

Figure 5.19 shows the comparison of the acquisition times for the variation of \( T_{int} \) and \( K \), normalized to the time it takes to perform the acquisition using 1 ms of data.

From Figure 5.19, it is verified that an increase in the coherent integration interval increases the acquisition time exponentially, unlike the increase in the number of non-coherent combinations which increases the acquisition time linearly. This difference is that, when the data used has length higher than 4 ms, the acquisition time is much slower when coherent combining is used, as evidenced in Figure 5.19. This difference in time can be justified by implementation of the algorithm, which has to perform the correlation several times corresponding to each branch, and to the increased number of data, which increases the length of the Fourier transforms.

Additionally, it can be seen from Figure 5.19 that executing the coherent combining algorithm until the secondary code chip position can be retrieved, unless high coherent integration times are necessary, is...
Figure 5.19: Acquisition time comparison between coherent and non-coherent combining.

not a good practice since it makes the acquisition slower. Therefore, for a fast acquisition, the secondary code acquisition should be performed using a more conventional method, as described in section 4.6.
Chapter 6

Conclusions

This chapter presents a summary of the conclusions reached along the work and some brief recommendation for the future work.

6.1 Conclusions

The main goal of this thesis, as stated in chapter 1, is to perform a complete study of the AltBOC modulation and all its components and to find software methods to achieve signal acquisition for the satellite signals present in the Galileo E5 band.

In chapter 2, the full study of the modulation was performed. There, it was found that the AltBOC modulation brings many upsides but also presents many challenges in the acquisition process, such as the complexity in implementation and multi-peak auto-correlation function, that make performing the acquisition in a short time a very challenging task.

To be able to capture the satellite sent signals, chapter 2.3 presents the most suitable configurations of hardware receivers containing the indispensable hardware components to make the software acquisition possible. It was concluded that there are multiple software configurations, each one with advantages and disadvantages. It was found that the Single Band receiver is the most suitable for fast acquisition on simple applications and that the Coherent Double Band receiver is the one that can achieve the best results, which is the most suitable for high precision applications.

Chapters 3 and 4 present the steps it takes to perform a correct acquisition. The conclusion here is that the acquisition is a complex process that can be performed in a variety of ways, each one presenting different results. In particular, the search space can be searched in a serial way or the search can be parallelized to achieve faster times. Additionally, the AltBOC modulation allows the acquisition to be performed not only for the full modulation but also for isolated components in order to achieve a faster speed.

In chapter 5, an analysis is done in terms of the acquisition time and the probability of correct detection for different C/N0 for the algorithms described in chapter 4. The major conclusions in this chapter were:
• Every method analyzed is capable to correctly perform the acquisition for the reference C/N\text{0} (46 dB.Hz).

• The difference in sensitivity is negligible when comparing the different algorithms. The major difference resides on which signals are acquired, being the direct acquisition the clear winner.

• The time of acquisition changes drastically with the algorithm used. The faster algorithms are the ones which parallelize the search space. In particular, the faster one is the DBZPTI which performs a two-dimension parallelization. There is also a big increase in acquisition time in the direct acquisition when compared to the SSB and DSB acquisition.

• The coherent combination produces better results compared to the non-coherent combination for the same data duration. However, due to the signal characteristics, the time it takes to perform the coherent integration, specially for large integration durations, is also much higher than the time needed for the non-coherent integration.

• Performing the secondary code acquisition simultaneously with the primary code acquisition is not a good practice (unless the signal requires the use of large coherent integration times), since it leads to a considerable increase in the acquisition time. Therefore, the secondary code acquisition should be performed after the primary code is already acquired.

To sum it up, the DBZPTI algorithm was found to be the most appropriate algorithm since it presents a sensitivity comparable with the other algorithms, but at the same time exhibits time results unmatched by them. Additionally, it was found that the SSB acquisition is the fastest but presented the worst sensitivity, therefore, it is suitable for fast applications where good carrier to noise ratios are expected. On the other end, it was concluded that the direct acquisition presents the best sensitivity results in exchange for a big increase in time spent; therefore it should only be used when the noise levels are expected to be high or when the SSB is not capable of detecting the incoming signals.

6.2 Future Work

An idea for a future work is to test the algorithms and methods implemented in this thesis using real data instead of simulated data. Even though there was an effort to emulate the characteristics of real data captured by a real receiver, some important issues were neglected, such as the finite number of bits in the analog-to-digital converter and the existence of error sources that can quickly vary the noise values of the signals. Only after being tested by real data that it is safe to say that the proposed methods are suitable to be used in day-to-day applications.
Bibliography


